DOCUMENT RESUME

ED 263 875 IR 011 859

AUTHOR Anderson, Gary C.
TITLE The Application of Computers in Liberal Arts Instruction. A Computer Literacy Faculty Development Project.
INSTITUTION Hamline Univ., St. Paul, Minn.
SPONS AGENCY EXXON Education Foundation, New York, N.Y.
PUB DATE Aug 85
NOTE 62p.
PUB TYPE Reports - Descriptive (141)

EDRS PRICE MF01/PC03 Plus Postage.
DESCRIPTORS *College Faculty; *Computer Literacy; Computer Software; *Faculty Development; Higher Education; Inservice Teacher Education; *Liberal Arts; Microcomputers; Peer Teaching; Program Descriptions; Teacher Improvement
IDENTIFIERS *Computer Uses in Education; Hamline University MN

ABSTRACT

In January 1983, Hamline University received a grant from the Exxon Corporation for a computer literacy faculty development project which involved three main activities: faculty workshops, faculty development projects, and hardware/software/courseware acquisitions. A series of four faculty workshops were offered during the 2-year grant period; most presenters were the recipients of Exxon grant stipends for faculty development projects, but some other knowledgeable faculty also made presentations. The last two workshops provided an opportunity for "hands-on" computer experiences on a variety of available computer systems. Approximately half of the grant monies funded 16 faculty members in a variety of academic fields (science and non-science) for projects designed with the objectives of developing faculty (and hence, student) awareness, skill, and knowledge in the application of computers in the classroom. A description of each of these 16 projects, including necessary hardware and software, is provided. (JB)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Gary C. Anderson

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

A B S T R A C T

The Hamline University Computer Literacy Faculty Development Project, sponsored by a grant from the Exxon Education Foundation, consisted of three main activities: faculty workshops; faculty development projects; and hardware / software / courseware acquisition. These activities had as their goal the increased incidental and integral use of computers across the undergraduate curriculum. This document contains reports on each of these activities and is thus a "history" of the project rather than a set of prescriptions for the application of computers in liberal arts instruction. These reports highlight the fact that faculty teaching faculty was a key concept in the project.

HAMLINE UNIVERSITY

THE APPLICATION OF COMPUTERS IN LIBERAL ARTS INSTRUCTION

I. INTRODUCTION

1. Hamline University

Hamline University is a coeducational, private, independent institution located in St. Paul, Minnesota and is affiliated with the United Methodist Church. As Minnesota's first college, founded by the territorial legislature in 1854, Hamline is deeply rooted in the history of Minnesota. This heritage of being first has characterized Hamline throughout its history as an innovator in higher education. Hamline was among the first colleges in the nation to require the demonstration of computer literacy as a graduation requirement. Pioneering efforts at Hamine have involved independent and individualized study, programmed learning, performance-directed teacher education, labs open 24 hours a day, special sections of regular courses, and other methods and concepts that give an extra dimension to education.

The College of Liberal Arts offers 33 undergraduate majors, operates on a 4-1-4 calendar, and confers the Bachelor of Arts degree. The School of Law, founded in 1972 and formally merged into the University in 1976, operates on a semester basis and confers the Juris Doctor degree. In addition, the University offers graduate courses leading to the Master of Arts in Liberal Studies and the Master of Arts in Public Administration.

Hamline has 85 full-time undergraduate faculty. More than 90 percent of this faculty hold doctorates or terminal degrees in their field. All of them teach in the classroom and serve as individual advisors; the student-faculty ratio is 13.5:1. The student population includes 1,200 undergraduates, 500 law and 150 graduate students. More than 3,000 people annually attend continuing education courses at Hamline.

Hamline is located in the heart of Minnesota's Twin Cities, midway between the city centers of Minneapolis and St. Paul, and is part of a metropolitan area of 2 million people. Seven colleges and universities, including the University of Minnesota, are within a 5-mile radius. Hamline's 32-acre campus is a comfortable, attractive blend of traditional and contemporary architecture in its nine academic buldings and six residence halls.

2. Computer Literacy At Hamline

In the Fall of 1980 Hamline University formed a "Computer Task Force" consisting of seven faculty members from the College of

Liberal Arts and representing the following disciplines: Economics, Education, Library Science, Mathematics, Modern Languages, Physics, Political Science. This task force formulated three tasks which related to various aspects of "computer literacy" at the undergraduate level. Task I -- to establish a general education computer literacy graduation requirement for the College of Liberal Arts of Hamline University. Task II -- to request a commitment from Administration and Faculty so as to increase and broaden the integral and incidental use of computers in all disciplines. Task III -- to introduce a Computer Science Program at Hamline which would be based on current resources and to initiate Computer Science as an allied field (minor).

A general education computer literacy graduation requirement for the College of Liberal Arts of Hamline University was passed by that faculty in November of 1980. This requirement took effect for Freshmen students in the Fall of 1982 and for transfer students in the Fall of 1983.

Hamline's "Computer Literacy Project" was based (in part) on several broadly defined (informational) dimensions of computer literacy as articulated in a Minnesota Educational Computing Consortium (MECC) report. This study specified the following informational objectives which can be used for the organization of instructional programs.

First, there is the hardware dimension which involves some understanding of the major components of a computer system and their function along with their historical development. A second dimension is that of software and data processing which often includes knowledge of how data is processed by computers using programs written by people and stored (with data) in computer memory. The third dimension is that of programming and algorithms which stresses the ability to follow, modify, correct, develop, and implement algorithms (expressed in English language statements, flowcharts, or computer programs). Knowledge of when and where computers are being used and what makes computer use suitable or unsuitable is an important aspect of the fourth dimension, applications of computers in society. The fifth dimension is the impact (positive and/or negative) of computers on society. This is different from the fourth dimension in that it deals with the effects or results of applying computers and needs to address many contemporary social issues. The sixth and last dimension deals with specific attitudes, values, and motivation; items which are affective rather than cognitive.

The computer literacy graduation requirement at Hamline may be satisfied in three ways: (1) Completion of an appropriate course at Hamline or another undergraduate institution which covers the dimensions of computer literacy. (2) Completion (at HU) of the course CS 100 - Computer Fundamentals. (3) Credit for documented previous experience with computers and computing, exposure to the various social issues in computing (via directed readings and/or CAI), and passing a proficiency examination.

Computer Fundamentals (CS 100) is a 1/2 (term) credit course that

has been designed to cover the dimensions of computer literacy. It contains three components: Introduction to Computers, Applications of Computers, Computers and Society. At the present time, students need to complete two of the following computer application (hands-on) units: BASIC computer programming, word processing, computer graphics, electronic spreadsheets, data base management. It is also possible to take this course on a self-paced (PSI) basis.

This manual, however, is directed towards Task II which concerns the next "level" of computer literacy -- the increased use (integral and/or incidental) of computers across the curriculum. We are now in the process of revising our curriculum to include, among other things, the development of computing intensive courses.

3. The Computer Literacy Faculty Development Project

In January 1983, Hamline received a grant from the Exxon Education Foundation for a Computer Literacy Faculty Development Project covering the 1983-84 and 1984-85 academic years. This project involved three main activities: faculty workshops; faculty development projects; and hardware/software/courseware acquisition. This manual contains reports on each of the main activities and thus is a "history" of the project rather than a set of prescriptions for the application of computers in liberal arts instruction. Faculty teaching faculty is one of the key development concepts used in this project.

A 1984 study of "Academic Computing Projects at Six Liberal Arts Colleges" by Andrew T. Masland of the Center for the Study of Higher Education at the Pennsylvania State University included the Hamline University project noted above. The Penn State report described the proposals and examined the colleges' progress to date. This report also discussed common resources the colleges needed to support academic computing.

On behalf of Hamline University, I would like to thank the Exxon Education Foundation and its Research Director, Richard R. Johnson, for their interest in and support of this project. I also would like to thank my many faculty colleagues at Hamline for their ideas, patience, and cooperation in the preparation of this manual. Comments, questions, and suggestions may be sent directly to the faculty members specified in each section of this report. It is our hope that each reader will be able to obtain some helpful information from this publication.

Gary C. Anderson
Professor of Mathematics
[Project Director]

II. FACULTY WORKSHOPS

A series of four faculty workshops was offered during the two-year grant period. The workshops were entitled "Computing As A Matter Of Course - x" where x = I, II, III, IV. The Fall term workshops were held in the morning and afternoon of a mid-term "break" day while the Spring term workshops were held in the afternoon and evening. These workshops provided an opportunity for all faculty members to learn about computer use at Hamline in a variety of situations, to discuss the problems and pitfalls of such use, and to give faculty people "hands-on" experience with the computer systems at Hamline.

Each workshop began with "coffee, donuts, and conversation" followed by a series of faculty presentations. Most presenters were the recipients of Exxon grant stipends for faculty development projects. Other presenters were the Project Director, the Director of Academic Computing, and faculty members who were already quite involved with the classroom use of computers. A modest honorarium was paid to the members of the latter group. A complimentary buffet luncheon or dinner divided the two series of presentations and provided an inducement for workshop attendees. Each workshop concluded with additional faculty presentations.

Workshops I and II made use of evaluation forms that asked questions about the expectations of each person in attendance, most and least useful sessions, suggestions for the next workshop, effectiveness, etc. Workshops II and III provided an opportunity for "hands-on" computer experience on a variety of HU computer systems.

Prior to each workshop all faculty members and selected administrators received an invitation and agenda/reservation form. The reservation form asked each person invited to indicate an intention to attend the presentation sessions, what "hands-on" session was requested, and interest in being a luncheon/dinner guest. Those people selecting a specific "hands-on" session were sent a short memo thanking them for their interest and telling them where and when their session was to be held.

The Project Director contacted faculty presenters in advance as to their availability for a particular workshop and then sent them a memo in which they were given some general guidelines for their presentation. These guidelines included some details as to the individual project motivation, what the presenter proposed, progress to date, and what lessons can be drawn from the project. Acceptance of a faculty development project stipend carried the obligation to make a workshop presentation and to provide a report to be included in this publication. The memo also requested what AV and/or computing equipment the presentation required. Presentors were also asked to provide a "nifty" title for their (30 minute) presentation. Although "nifty" is a somewhat ambiguous term, here is a small sampling of such titles: "Computing at Hamline and the Lessons of Sisyphus," "Cost/Benefit Paradigms Applied to Biomedical Ethics,"

"Inexpensive Instrument Interfacing for Instilling Interest," "PLATO in Palestine," "Microcomputer Modeling in Biology: Life on the Screen," "ALICE in LogicLand," and "Buddy, Can You Paradigm?"

Since most of the faculty workshop presentations/reports will be found in a later section of this manual, we only include here a special presentation "Computer Text-Editing: Examining the Trojan Horse" given at Workshop II (4/5/84) by Ms. Linda Hunter, Director and Verbal Skills Coordinator of the Academic Support Center and an English Instructor at St. Olaf College, Northfield, MN. Also included is her "A Beginning Bibliography for Computers and Composition."

"Computer Text-Editing: Examining the Trojan Horse"

Linda Hunter

When Gary asked me for a nifty title for this presentation, I was definitely pressed. Earlier versions of this talk were entitled "Basic Writers and the Computer" and "Composition and the Computer" - hardly nifty. My first attempt at a semi-nifty title for this talk was "Computer Text-Editing: Not a Trojan Horse." I shared this idea with a colleague in Mathematics who smiled and said, "But it is!"

I had been considering text-editing from the view of a humanities faculty who suspect that installing computers in a liberal arts environment is just the first step on the road to selling out to technology. He, on the other hand, was thinking of a Mathematics faculty that is attached to the computer for its number-crunching capabilities whose realm is invaded by text-editing in other fields. So we decided that examining the horse might be a more appropriate metaphor.

Before I begin that examination, I think you should know a bit more about me. As an English major, initially inspired by George Vane at Hamline (from whom I contrived to get six courses before I transferred at the end of my sophomore year), I want to assure you that I am not a closet mathematician. If you doubt me, you should consult Dr. Kay Smith of our Mathematics Department who coached me through our basic programming course. Neither do I like machines. I'm the type who always looked for a student to run a film projector. I didn't even take Gary up on his offer of an overhead. I'm sure it wouldn't work for me. With that background, you can be sure that text-editing is wonderful, or I wouldn't give it a second look.

It was not altogether by choice that I first got involved since it was a condition of employment when I went to St. Olaf. The then director of the Academic Support Center asked if I'd be comfortable teaching my Writing Place tutors to assist in text-editing. I

ignored the knot in my stomach and cheerfully said, "of course!" Developing a now archaic text-editing handbook for my tutors was one of my first tasks. It is a good illustration of the of change in computing. That handbook instructed students in using the line-oriented editor on our mainframe computer. Within three months it was outdated by the cursor-oriented system on the micros. And we've gone on to a third alternative, cursor-oriented editing on the mainframe since then.

Now to examine the horse, I'll explain a bit about computing at St. Olaf. Those in charge of decision-making about the direction of computing at St. Olaf decided to go with Heathkits, H-89's, both as micros and terminals three or four years ago. In the last two years two clusters have been established of 8-12 computers in a room, each with two disk drives and hook-ups to a printer and the mainframes, a PDP-11/70 and a VAX. These clusters were not constructed just for text-editing, but that is probably their main use. There are about 90 total computers on campus for public use for a student body of about 2,900. About two-thirds of our students have active accounts on the system.

The text-editing programs we use are PIE and a formatter revised from a commercial program, TEXT, which we call RUNOFF. I understand from my son at Hamline that your RUNOFF program on the VAX is very similar. This means that students use one program for editing and run their file with the embedded commands through another for formatting. The theory behind that decision is that students needn't concern themselves with the "pretty page" until they are ready to run it off. This concern can severely hamper developing writers, as Mina Shaughnessey's studies recorded in Errors and Expectations showed. Students need only three or four commands to format it in lovely double-spaced, typed form, complete with centered headings. With a few more they can do endnotes, underlining, quotations, and other niceties.

The first students for whom I planned to require text-editing were in a developmental writing course, English 10, called Critical Skills in Composition. The Director of Writing selected 100 students whose aptitude scores and high school grades suggested they might have difficulty in freshman English for an hour long essay test during Freshman Week. The essays were graded holistically by the English department members. The bottom 45 were chosen for English 10. They did not feel chosen, as you might suspect -- sentenced would be more like it. Attitude toward the course was clearly a problem.

You might wonder why I wanted to add the task of using the computer to the task of learning to write effectively for these students. The philosophy behind the structure of the course was that revision was the heart of writing. On Mondays we gave assignments and expounded theory. On Wednesdays students brought rough drafts to share with their small groups. On Thursdays or Fridays students conferred with us on a revised draft in light of the insights they got from their small groups. The following Monday students turned in yet another draft, revised on the basis of their conferences with us. This structure, it seemed to me, begged for

the computer to make all those revisions simpler. I also hoped that a new tool would promote a new attitude. If I had been wrong, no doubt I wouldn't be here.

I integrated the computer from my syllabus to the course-end evaluation. In fact, I listed the Computing Center documentation as a text on the syllabus. However, I decided I couldn't afford a lot of class time devoted to computer instruction. For English 10 students I have a one-hour orientation in one of the campus computer rooms in which I show them the basics: getting in and out, moving the cursor around, and deleting and inserting options, plus the most basic formatting commands. One fellow followed me back to my office that first year, telling me that he'd never be able to work on the computer, exhibiting all the signs of computer anxiety. I offered to whip up a one-page quick and dirty handout to get him started, and I've used it ever since. He taught me the first of many lessons in this experiment: students are not going to read the documentation like a text, even if I list it that way. Oral tradition and experimentation is the preferred method. Now I have a pool of ex-students who are delighted to share their prowess with initiates. There has also been some talk of assigning a knowledgeable student to a department to assist students in handling questions about particular text-editing programs such as our new one that will handle foreign languages.

Only when all else fails will students turn to the documentation. However, when they do, it helps to have clear, well-written documents. Unfortunately, many computer programs have nearly unintelligible directions and explanations. Our Computing Center staff rewrote part of our commercial programs and consequently produced much more readable prose. The documents also have quick reference guides at the back which I suspect are much more used than the rest of the manuals. Copies of all documentation are placed in the computer rooms around campus in case students don't have their copies with them.

Happily, I don't have to maintain my computer expert pose very long because class experts quickly emerge, although at least half of the classes had no prior experience with computers. When a student tells me in a conference about her success with the Search capability of the computer to find errors, I ask her to share this good news with the class in the last five minutes. (A parenthetical note on that capability: One of our seniors did a senior distinction paper on Spenser, spelling the name with a "c" throughout. She didn't discover this potential disaster until a day or two before it was due. The computer repaired all the damage within two minutes. If she'd had to type the entire 30 page paper over, that might have been one distinction paper left unfinished.) In addition to class experts, the Computing Center staffs the clusters with consultants during the busiest times. A feeling of community also quickly arises in clusters because the person at the next computer might well have experienced the same problem you're having, and is nearly always willing to help.

Students rarely return to their graduation typewriters after doing a paper or two on the computer unless desperate. It isn't too

surprising that developmental writers are willing to use the computer, since their earlier systems clearly did not work. Students not only write drafts, but even prewriting and journals on the computer. All the students I've taught but one, a recalcitrant music major who preferred guitars to keyboard instruments, felt I should continue to require the computers. Nearly all taught other students to text-edit, and used it themselves in other courses. I made the mistake on my first course evaluation of asking whether the small groups, conferences with me, or the computer was the strongest motivator for revision. Sadly enough, three students found the computer the strongest! I haven't asked the question since.

The desperation that I referred to comes from three main sources: the level of demand, mechanical problems, and personal time management. In some cases the lines to use the computers get long, especially in the late afternoon and at night. It takes a while for the obvious lesson to be learned: use the computers during the lunch, dinner, or morning hours. Mechanical problems do arise with disk drives being down and disks that malfunction. However, the most frequent problem is simply time management. If the paper isn't ready to be typed until 4:00 AM on the day it is due, regrettably, the terminals are not available. We'll fix that problem next year, since we plan to locate two small clusters in dorms. However, it is a problem this year. My students quickly learn I have no sympathy with it.

Now you may be convinced that students like computers, but what about their writing? The first year I was the only one of the three instructors in English 10 to require text-editing. When we did our group grading at mid-term time, my colleagues were amazed by my students' fluency. I take little credit for it, since we were using the same format, but their papers were longer and more developed. This is a typical result of text-editing at a variety of grade levels. Fluency is not a measure of excellence, however. Skill in revision may at least suggest some new competencies.

One of the characteristics of developmental or basic writers, as you may know, is preoccupation with revision on the word or sentence level. It is a bit painful to watch one of these writers begin a draft. As Mina Shaunessey notes, these students cross out almost as many words as they write on that first draft. With the computer, they not only create more text, they find the task of revising it much more palatable.

That reminds me of a story Hugh Kenner, a Joyce scholar, told about Joyce when Kenner was at St. Olaf last year, to speak about computers. Joyce was working on <u>Ulysses</u> in Zurich, and one day after a long day's work, his friend, Frank Martin, encountered him.

"Been working today?", Martin asked.

"Yes."

"All day?"

"Yes."

"Get much done?"

"Two sentences."

Martin, in astonishment, asked "Were you looking for the exact word?"

Joyce replied, "Oh no, I had all the words. I was looking for the right order in which to put them."

The computer is a wonderful instrument on which to play with those words. Joyce, doubtlesss, would have been amazed.

Even basic writers discover that quickly and go beyond word level to revise. Computers take the fear out of that first draft. This lack of fear assumes ease of revision, an attitude not characteristic of basic writers (and many other writers, I might add.) In comparing sequential drafts from my students with my colleagues, we could see that my students made larger, more substantive revisions, moving paragraphs, changing order. This pattern of revision is generally identified in composition research as the type made by good writers. This willingness to revise was our goal, and the computer seems to have been an important tool in achieving it. However, I don't think that my students would have been so likely to make those revisions without human encouragement.

I used to tell my students that they would save time with the computer. I no longer tell them that. I imagine that time will always be a factor in any of our writing. The difference that the computer makes is in the quality of the time, since it can be spent on higher level considerations.

This brings me to another use of computers in composition that I've been helping to pilot this semester. Three of us at different levels of writing classes, Freshman English, Expository Writing, and Technical and Scientific Writing, are introducing our students to Writers' Workbench, a Bell Labs program for analyzing prose. Not only will it print out misspelled words, but it will flag wordy phrases, abstract language, and sexist language. It also gives a count of sentence types - simple, compound, and complex; the passive verbs, nominalizations, split infinitives, and expletives. I require a Writers' Workbench printout with papers that are to be graded, but I discourage premature use of the program for rough drafts. That would simply serve to distract students from the task appropriate to that stage of their writing process -- clarifying meaning and organization. At this point I suspect that its novelty is contributing to my students' excitment about it. I do point out that positive print-outs of Writers' Workbench do not guarantee A's, but my comments and grading are enormously simplified by their using it. I can direct my comments to the important elements in logic, content, cohesiveness rather than the editing problems.

I can also get their attention on matters that surprised me. One student called my office on a Friday afternoon, urgently requesting an appointment. "Writers' Workbench tells me that I have too many passives, and I can't find them. I've got to find out what's

wrong!" Never have I had a student ask for a lesson on the passive.

In using the computer with regular freshman English students this semester, I've observed even faster conversions from the typewriter, but that may be partly because they chose my section, clearly described as using the computer. I did not even devote an hour to orienting students to the system, and got few complaints. English 11 students are now also back on the mainframe with a revised version of PIE because the micros got too overcrowded, and the VAX had room for more use. That has the added advantage that students can use electronic mail to communicate with me between classes, helpful if problems appear, and more efficient than phones many times.

Yes, there are some snakes in this paradise. The mechanical ones are perhaps the most frustrating. Some of our bargain disk drives simply did not hold up in the public micro rooms. The disks themselves provided some problems, as I mentioned earlier, since setting them on magnetic stereo speakers or exposing them to prolonged cold tends to disable them, at least temporarily. The mechanical problems quickly become psychological when the error message "Fatal System Error 002," or "File Dumped to Core," flashes across the screen. One student recorded his feelings in his journal, suggesting that he was motivated to physical attack on the computer, only being restrained by others' presences. Being back on the mainframes has solved the disk and drive problems, for the time being. But as long as machines are fallible, there will be that kind of frustration. Prolonged use tends to reduce it.

The third type of problem is capsulized by one of my colleagues who contends that the malfunctioning computer has replaced the grandmother's funeral as the most popular excuse for not having a paper ready on time. I experienced an interesting variation on this a couple of weeks ago when we had a minor flood in the computer room. This caused the machines to be down all day Friday. Now my students' weekly papers are due on Tuesday. I must have gotten at least five inquiries if their papers would still be due on Tuesday since they couldn't use the computers on Friday! I pulled two positive messages out of this experience. My students were starting their papers early, and they really wanted to use those computers! I suggested that if the computers really were down all weekend, earlier technologies were still available.

Computer text-editing has other side benefits related to better writing. It promotes precision as well as objectivity. Errors in commands simply are not tolerated. A semi-colon instead of a colon produces no action. A space in the wrong place gets an error message. I think this same precision has some carryover in editing. Computers also promote openess to reader response since the changes can be made so easily, an important bonus, since many students have a tendency to defend their first choices. Rather than being cast in concrete, papers are written in a light, a much more fluid medium.

In examining the Trojan horse, I found its danger minimal. I believe that it is true primarily because of the environment we have created for its use. We have carefully provided for considerable

human input into students' writing within the course structure, and considerable human support to their computer use. Some of you may be aware of the other studies that are less sanguine than my informal one. I am convinced that the results have been skewed by difficult text-editing programs or the lack of human intervention in the process. An interesting study is being done at the University of Minnesota as Lillian Bridwell and Donald Ross contrast the text-editing of experienced and inexperienced writers, using a program that records every key stroke on the computer, giving graphic evidence of the process. Their study should give us some more helpful information.

Perhaps another metaphor will clarify the role of the computer in writing. Hugh Kenner suggested that the computer screen can mirror our mind's eye in a new way, fostering creativity in ways we'd hardly expect from a machine. We can get down ideas almost as fast as they occur to us. Then we can also examine those images critically. That two-fold advantage may well make unforeseen changes in writing. I suggest that the computer is not a panacea for writers, but it is a significant tool in the writing process that we should make widely available.

"A Beginning Bibliography for Computers and Composition"

Linda Hunter

Barth, R. "ERIC/RCS Report: An Annotated Bibliography of Readings for the Computer Novice and the English Teacher." English Journal 60 (January, 1979).

Bell, Fred. "Classroom Computers: Beyond the 3R's." Creative Computing 5:8 in Computer Connections for Gifted Children and Youth. ed. Jean N. Nazzaro. ERIC, ED209-899. 1981.

Bloch, Stephen. "Reflections of a Computer Language Nut," in Computer Connections for Gifted Children and Youth, ed. Jean N. Nazzaro. ERIC, ED209-889. 1981.

This essay is by a gifted teen-age user of computers, arguing for exposure to computer programming.

Bradley, Virginia N. "Improving Students' Writing with Microcomputers." Language Arts 59:7 (October 1982), 732-743.

Includes two studies of elementary students using Apple Writer, Easy Writer, and Super Text. Draws some conclusions about necessary qualities in a word processing program for elementary schools.

Burns, Hugh. "Computers As a Way of Brainstorming." Proceedings

of the National Educational Computing Conference. 1981, 105-108.

--------, and George Culp. "Stimulating Invention in English Composition through CAI." Educational Technology 20 (August, 1980), pp. 5-10.

Caldwell, Robert. "Guidelines for Developing Basic Skills Instructional Materials for Use with Microcomputer Technology." Educational Technology 20 (October, 1980).

Collier, Richard M. "Computerized Word-Processing as an Aid to Revision." College Composition and Communication 34:2 (May 1983), 149-155.

Computers in the English Classroom. A Primer for Teachers. Sally N. Standiford, Kathleen Jaycox, and Anne Auten, ed. NCTE, 1983.

This essay collection really is a primer, an elementary introduction to computers in the English classroom, including a glossary of basic computer terms.

Computers in Teaching English. James E. Davis and Hazel K. Davis, ed. Spring 1983 Focus of the Southeastern Council of Teachers of English. NCTE, 1984.

This collection contains twenty-five articles covering computer use for teaching basic writing, technical writing, nontraditional students, and other applications. It ends with a selected, annotated list of resources for computer use in English instruction.

Connel, Bruce and Ann Humes. "Using Microcomputers for Composition Instruction." ED 203 872. 1981.

Daiute, Colleen, et.al. "The Computer in the Writing Class: Problems and Potentials." Proceedings of the National Educational Computing Conference 1983, Computer Society Press (1983), 22-26.

----------. "The Computer as Stylus and Audience." College Composition and Communication." 43:2 (May 1983), 134-145.

Evans, Christopher. The Micro Millenium. New York, Viking Press, 1979.

Gabel, David. "Word Processing: Finding the Right Software." Personal Computing 7:4 (April 1983), 110-125+.

Kenner, Hugh. "A Professional Writer Looks at Budget Word Processing." Sextant. Spring, 1982.

Macdonald, Nina. "Pattern Matching and Language Analysis as Editing Supports." Bell Laboratories: Piscataway, NJ.

This paper describes the Bell Labs programs.

Marcus, Stephen. "The Muse and the Machine, A Computers and Poetry Project." Classroom Computer News 3:2 (November/December 1982) 28-31+

Monahan, Brian D. "Computing and Revising." English Journal. November 1982. 93-95.

This article briefly describes the author's use of TRS-80's in a high school English class for text editing.

Nold, Ellen. "Fear and Trembling: The Humanist Approaches the Computer." College Composition and Communication 25 (October, 1975).

Designed to allay our worst fears and point out creative possibilities.

Olds, Henry F., Jr. et. al. "Word Processing, How Will It Shape the Student Writer?" CCN Forum Classroom Computer News 3:2 (November/December 1982) 24-27+.

Piper, Karen. "Word Processing in the Classroom: Using Microcomputer-Delivered Sentence Combining Exercises with Elementary Students." Proceedings of the National Educational Computing Conference 1983. Computer Society Press, (1983), 18-21.

Riordon, Tim. "Regarding that Request for a Microcomputer." Classroom Computer News 3:2 (November/December 1982) 37-45+.

Rodrigues, Raymond J. and Dawn Rodrigues. "Computer-Based Invention: Its Place and Potential." College Composition and Communication, 35 (February 1984), 78-87.

Roth, Audrey. "The Computer in the Kitchen." 14 pp. ED 140 284 EDRS MF .83 HC $1.67 plus postage.

A brief survey of technological possibilities in the area of English.

Rubin, Andee. "The Computer Confronts Language Arts: Cans and Shoulds for Education," to appear in A. C. Wilkinson (ed.) Classroom Computers and Cognitive Science. New York, Academic Press, 1982.

Schwartz, Helen. "Monsters and Mentors: Computer Applications for Humanistic Education." College English 44 (February, 1982).

Schwartz, Helen and Lillian S. Bridwell. "A Selected Bibliography on Computers and Composition." College Composition and Communication, 35 (February 1984), 71-77.

Sokoloff, Michele. "Word Processors: A Look at Five Options." Media and Methods 18:6 (February, 1982) 25+.

Deals with word processing on APPLE, TRS 80 and PET microcomputers.

Steinberg, Esther R. ed. "Critical Incidents in the Evolution of Computer Projects." University of Illinois, 1977. ED 148 298 EDRS MF $.83 HC $1.67.

Points out successes and failures in the history of PLATO, an extensive computer project in education in Illinois.

Wall, Shavaun, M. and Nancy E. Taylor. "Using Interactive Computer Programs in Teaching Higher Conceptual Skills: An Approach to Instruction in Writing," Educational Technology 22 (February, 1982), 13-17.

Wresch, William. "Computers in English Class: Finally Beyond Grammar and Spelling Drills." College English 44:5 (September 1982), 483-490.

Wilson, K. "English Teachers: Key to Computer Literacy." English Journal 70 (September, 1980).

Urges English teachers not to follow Luddite examples!

Zoller, Peter. "Composition and the Computer." 7 pp. ED 127 611. 1975.

Describes an experimental use of computers in an writing workshop.

III. FACULTY DEVELOPMENT PROJECTS

Approximately half of the grant monies from the Exxon Education Foundation went to fund stipends to sixteen faculty members from a variety of academic fields (science and non-science). A five-person committee (chaired by the Project Director) consisting of faculty from the Chemistry, Economics, Education, and Religion departments served to coordinate the selection process. These individual faculty projects, typically conducted during the summers of 1983 and 1984, were to be designed with the objectives of developing faculty (and hence student) awareness, skill, and knowledge in the application of computers in the classroom.

The Project Director made information about this opportunity and application forms available to all College of Liberal Arts faculty early in the Spring terms (1983 and 1984). It was noted here that all stipend recipients would be required to participate in at least one workshop during the forthcoming academic year and to provide information on their projects for inclusion in this manual. The application forms inquired about an applicant's current level of computer competency (programming language proficiency, previous computer use, and the option: No computer use -- YET!) and academic or other time commitments already incurred for their project development period. A written project proposal was required that had to include information about what the faculty member proposed to do, the plan to proceed, which classes would be involved in the project, when these classes would be involved, what existing computing facilities would be used, and what software (existing or to be acquired) may be needed.

The Exxon Grant Committee then met, evaluated the applicants' project proposals, and selected the stipend recipients. In Spring 1983, six stipends were awarded for Summer 1983 and two for Summer 1984. In Spring 1984, eight stipends were awarded for Summer 1984. Each applicant promptly received a letter from the Committee with information about the committee decision. Grant recipients were again reminded of their obligations as to workshop participation and reports for this document. Those receiving negative decisions for 1983 were encouraged to reapply for 1984. All the applicants were informed that other Exxon grant funds were available for the acquisition of course-related software, programming and consulting assistance, and other support services.

The stipend recipients made presentations at one of the four faculty workshops. Early in the 1985 Spring term the first of three memos from the Project Director to stipend recipients (and others who received funding from the Exxon grant) alerted these faculty to the need of providing information for this manual. The second memo in early May provided another reminder and some general reporting guidelines. The final memo in early June reiterated the contents of the first two memos.

The following reports are then the result of the individual faculty development projects which have served to increase the

incidental and integral use of computers and computing across many disiplines at Hamline. Included in some reports are details relating to the acquisition and use of certain software packages that were part of the faculty development project. Also included are occasional comments from the Project Director (and editor). Please contact a specific faculty member for additional information relating to a certain project.

1. BIOLOGY DEPARTMENT - "Graphic Simulation of Biological Systems" - an intradepartmental workshop to infuse the use of microcomputer graphic simulations into a wide range of biology courses.

a) John Brennan - "Microcomputing in Biology"

In June of 1984 I conducted (for Biology Department colleagues) a workshop on using BASIC to simulate biological processes. Through several lectures, personal assistance, and James Spain's text, BASIC Microcomputer Models in Biology, colleagues learned to solve equations by numerical methods and to display the solutions graphically.

Colleagues performed initial exercises in disciplinary specialties of interest, including genetics, ecology, and biochemistry. On their own, they later ventured into problems requiring greater sophistication.

Colleagues have found it easier to purchase software (from the Exxon grant) rather than to invest the large amounts of time needed to conceive, create, and implement custom-tailored software for their courses. I have developed, and am continuing to develop, several software programs that enable students to interactively explore the dynamics of enzyme function and titration of weak acids for the physiology course, Biology 347 - Organismic Function.

The (January) Interim course, Microcomputing in Biology, given in 1984 and 1985, enabled students to gain basic programming skills to simulate quantitative models of biological processes. We also examined ways that the microcomputer may, through its interfaces with electronic instruments, process and display data from laboratory experiments.

Purchases and roles of acquired software/hardware --

1. New Step By Step: several students lacking previous programming skills used this package to learn BASIC programming.

2. Know Your Apple: used to orient students to Apple computers.

3. Basic and Graphics Workshop Manuals, Resources Handbook, Laboratory Instrumentation Manual, Computer Simulation texts, Physiological Simulations for the Apple: these resources provided

background readings for students as they planned their own projects.

4. Graphing and Modeling in BASIC (diskette): provided an alternative software program for students to explore graphics techniques.

5. Human-80 Comprehensive Physiological Model and the Microsoft Premium Softcard IIe: used to demonstrate how a powerful compiled program can simulate the integrated response of a complex organism to physiological stressors. This program is also used in the laboratory of Biology 347, a sophomore physiology course.

The Exxon grant enabled us to purchase valuable resources and tools that would have otherwise put a severe strain on a slender departmental budget. I earnestly hope that the institution secures support for future ventures in microcomputing.

b) William Downing - "Applications in Freshman Biology Laboratorty, Ecological Records, and Writing for Courses"

This paper is in two parts: proposed laboratory uses for graphics and tabulation; then, uses for, but not in, biology laboratory and lecture.

I have used my home Kaypro II in writing <u>for</u> the courses I teach at Hamline University for more than two years, and it is now second nature to use it: I have developed most of the "Perfect Writer" forms and headings I need, and it is easy to call up forms for laboratory directions, syllabi, study guides, and tests. My use <u>within</u> the laboratory is new, dating essentially from the summer of 1984, when the Biology Department added enough Apple IIe computers to make laboratory use feasible. This laboratory use will require several years of development before it, too, is second nature.

<u>I. PROPOSED LABORATORY USES FOR GRAPHICS</u>

Professor John Brennan designed a workshop for the Biology Faculty to give information concerning the library of commercial software provided in the Biology Department and to give guided practice in programming to produce appropriate graphics. The required text by Spain will be a continuing aid in programming for graphics, as I continue my practice. Fortunately the Department owns the software which I find most immediately useful in Freshman laboratory. See "A" below for details.

The requirements for tabulations of huge quantities of measurement data in ecological research recording are more complex, and will demand further self-instruction during the summer of 1985, using "AppleWorks" as far as it will go and augmenting from there -- see "B" below for details.

A. GRAPHING IN FRESHMAN BIOLOGY

Early in the term the instructor-written laboratory manual calls for a rate study of the action of the enzyme polyphenol oxidase, measuring with spectrophotometer the rate of production of brown-colored benzoquinone.

The protocol is as follows:

"The instructor will give your team a flask of filtered enzyme extract. This is the same kind of enzyme extract as made during the lecture demonstration.

"Mix three cuvette tubes as follows:

"Use the wavelength determined above. Measure immediately after mixing, and measure at 5-minute intervals for as long as you can observe any change in the experimental tube. Continue agitating the tubes so you can be sure there is an excess of oxygen all the way to the bottom. One person should remain in charge of this experiment, while the other team members go on to the next section.

"Record data below:

Once this experiment is begun, students carry out experiments based on it, using the same methods and similar graphs, varying amounts of particular ingredients, temperature, etc., measuring: effect of substrate concentration, effect of enzyme concentration, effect of temperature, the enzyme as a digestible protein, the enzyme as a denaturable protein, dependency on copper ions.

At present, students graph each experiment separately on forms provided, and compute the rate of enzyme action. The use of the computer directly in the laboratory will not only give computer graphics familiarity at the beginning of their college biological careers, but will provide an automatic rate formula at once.

2. Bacterial Growth Curve

Later in the same term of the beginning Freshman course, the students receive an introduction to bacteriology, and one of the experiments involves rate of growth of bacteria.

The protocol is as follows:

"When placed in a new environment rich in materials necessary for life, almost any living species will undergo a predictable "growth cycle."
We will study this in bacteria, using a spectrophotometer to indicate how many bacteria are present at any given time.

"We will expect to obtain a curve such as the following:

The students then grow bacteria in cuvette tubes and measure with a spectrophotometer the amount of light that passes through the cultures after having been scattered by the bacteria. They measure at regular intervals until the growth levels off.

Introduction of the computer in the laboratory will allow the students to enter the data in a program that will graph the data and automatically determine the doubling rate of the bacteria. The software for this experiment is now in the Department files.

B. TABULATION AND GRAPHING OF DATA IN ECOLOGICAL RESEARCH

Four students will be working with me on independent study during the summer of 1985, measuring growth rate of clams (Unionid mussels) in northeastern Minnesota lakes, under a research grant from the Blandin Foundation. We will correlate the growth rates with acidity in the lakes under the hypothesis that high acidity will slow or stop growth of the shells.

We will take one of the Department's Apple IIe computers with us to the laboratory of northeastern Minnesota where it will be indispensable in recording the large masses of data collected and dealing with them statistically. Present work has shown that the three programs provided in "AppleWorks", plus the graphics program described in "A-2" above, will suit our purposes very well.

II. USES FOR, BUT NOT IN, BIOLOGY LABORATORY AND LECTURE

For the other biology courses that I teach -- Vertebrate Histology, Comparative Vertebrate Anatomy, Human Anatomy, and Environmental Choices -- the computer does not lend itself readily to use "in the classroom" or "in the laboratory"; yet, I have found it to be indispensable in my teaching. I use my home computer -- Kaypro II -- daily, year-around, in preparing and teaching my courses, in producing laboratory manuals, scripts for slide-tape modules, study guides, syllabi, and tests. Introduction of the home computer into my life has improved my teaching, by expanding the scope and amount of work I can do. My workshop presentation to the Faculty included the following:

1. Laboratory Manual: I substantially rewrite this 200 page Hamline-produced book of laboratory directions for Biology 115 (Cell Structure and Function) annually to keep pace with the rapidly-changing field of Cell Biology; and now that it is on disks, the computer offers new flexibility in the revision process. Thus it has not only become easier, but it has made it possible to make relatively small changes on an annual basis that I would not have troubled to do in the past. One example will show the advantage: In the coming year I intend to change the animal microtechnique from "Necturus" to "frog" tissues. This will mean not only searching out and changing the word "Necturus" to "frog" within the laboratory assignment that takes up microtechnique, but in four other different chapters. It would necessitate a careful search through the written material in the manual to be sure that I did not miss reading and changing; then it would require re-typing of many pages before re-printing could be carried out again. With "Search and Replace"

in my Perfect Writer program, this long set of routine activities will be reduced to a few simple operations.

2. Slide Tape Modules: Under a Bush Foundation grant, I photographed, wrote scripts, and produced ten slide-tape modules for Biology 115 (Cell Structure and Function), showing methods of carrying out laboratory techniques. In addition I have made photomicrographs of cells and tissues for the course Biology 332 (Vertebrate Histology), and have put them into five carrousels (so far) with taped commentary for viewing by students. All tapes require scripts, and without the Perfect Writer program I would not have been able to do this rapidly and have as many complete as I do now. The footnote system in Perfect Writer provides the best means possible for coordinating a taped script with slides because it provides at one sitting, a numbered sequence of cues to push the forward key of the Caramate projector to key for automatic advance and the description of the slide at the bottom of the page. If I find I have omitted a slide or want to change sequence, I can simply remove the footnote, and the software will automatically change the numerical sequence. No other office aid would be as useful, easy, and flexible, as a writing program like Perfect Writer; and thus I feel justified in saying that the computer is not only helpful, but indispensable.

3. Study Guides: A similar program to the above, "Enumerate", will enumerate paragraphs automatically. As I write, I do not know what number will be assigned to a paragraph, but when it is printed, it will be assigned a number. If I decide to eliminate this one, or move it to another place, all members of the set will receive new numbers. Thus, when reading the text assigned to the students in any course, I can identify items I want them to emphasize, point it out in a question, a reference, or a statement; and when I finally have the questions in the sequence I want, I print out and the paragraphs will be in numerical order. This has aided my teaching, and it is a process I would not use without the personal computer.

4. Syllabi: In the past these have presented a problem, because, although changes from year to year are often minor, they have required complete revision and re-typing. The content of the schedule changes from year to year; the sequence of lecture and laboratory assignments change from year to year; and the calendar shifts enough so that every entry requires change. With the computer, once a schedule in a course syllabus has been established, it is almost an automatic process to modify the content within weeks, as well as to alter the dates, using Pefect Writer. As above, it is unlikely that I would undertake this kind of complex change unless I could do it with the word processor.

5. Tests: Multiple choice questions, particularly, are wonderfully easy to write with a word processor. In many courses I use an immediate feedback type of response sheet, which requires placement of the correct responses in (a), (b), (c), or (d). Marking the proper one ahead of time, e.g., ($b), shows me where to put the answer, and then with one sweep I can remove all the $'s before printing. Every instructor should enjoy the ease of such writing!

My use of the personal computer for, but not necessarily in, my courses, has expanded my abilities beyond anything I would have imagined. I recommend that every instructor have such a tool as standard office equipment.

c) Sylvia Kerr - "Microcomputing in Biology"

During June 1984, John Brennan ran a three-day workshop for Biology faculty. During this workshop we used the text BASIC Microcomputer Models in Biology by James Spain as a guide to the construction of short programs that allowed us to input experimental variables and plot rate diagrams by chaining our programs to a pre-existing plotting program.

I feel that the three-day workshop and the week I spent getting ready for it were very valuable experiences. Previously, I have used our home microcomputer extensively for word-processing and preparation of course materials. I had also used commercially available software as a part of student laboratory exercises. But I had not "taken the plunge" and tried programming on my own.

I learned several things. First, I have a much better idea of what happens in computer courses. Second -- the experience I had was positive. (Frankly, I have very little time while at Hamline to devote to developing programming skills, and my previous attempts to access the VAX and the former PLATO system were discouraging if not enraging.) Thirdly, I learned that I actually enjoy programming. In the past, my favorite part of mathematics was the construction and improvement of geometric proofs, and writing simple programs reminded me of this previous experience. As a result of the workshop I know that I can program if I set aside the required time. And since I am now somewhat acquainted with simple programming techniques -- it is easy to think of projects that are worth pursuing.

Unfortunately, I have done almost nothing to further my programming skills since the workshop. The fact that our home micro is a Radio-Shack and not an Apple contributes somewhat to the difficulty because I would want to write programs for students to use and they only have access to Apples. Discs and languages for the two systems are incompatible, and Hamline lacks modems. In addition, I devoted my summer and my Fall term sabbatical leave to another project that was in every sense of the word -- "full" time. Spring term was so busy that I couldn't "add on" another project. I do, however, intend to try to make changes in at least one program we own in time to use an improved version in a Fall term course.

The Exxon grant also allowed me to purchase "Micro-Micro" -- a quiz program on six discs (56 programs) geared to standard medical microbiology texts. At the time I reviewed the sampler disc, two students were enrolled in microbiology for Spring term, and I was considering an altered course format. By the time the course began, twelve students had enrolled, and conversion of the course from

lecture to independent study-microcomputer format was no longer warranted. I will definitely use individual programs from the software course in the Immunology course scheduled for next Fall and for individual exercises in other courses. And, if students request independent study or a PSI course in medical microbiology in the future, I can now structure this course.

d) Keith Klein - "Microcomputing in Biology"

My involvement in the Exxon grant was to attend a computer workshop on the use of Apple computers in Biology classes and to acquire some programs for evaluation and possible use in my classes. During the workshop we were introduced to the possibilities of the Apple computer as a model-building device and graphic display system. Several programs were provided as examples and included models of Kinetics, Ballistics, and Random-Number Generators. I used most of the time spent during the workshop to teach myself Applesoft BASIC. I was able to write two new programs that allow students to analyse gene frequencies in natural populations and compare data from several populations with another.

The programs I was able to purchase and evaluate were Flygen, Moths, and Popgen (all copyright Diversifed Educational Enterprises, Inc., 725 Main Street, Lafayette, IN, 47901). For several reasons none of these proved useful for my courses. The instructions that come with programs such as these are of generally poor quality. Apparently the authors are "computer freaks" who have little or no grasp of the niceties of the English language and speak a kind of jargon foreign to the average person. Also disturbing was the use of programming "gimmicks" such as color graphics with no relation to the program content. Color logos and flashy introductions are fine, but there should be color graphics in the <u>body</u> of the program too. All too often such introductions were followed by a text-only format which is very difficult to read on color monitors.

e) Ellis Wyatt - "Microcomputing and Ecology"

Population ecology utilizes the exponential growth equation that has been modified by inserting variables into it. This allows the generation of logistic or sigmoid growth curves under a wide range of limits for a given variable. Models that involve complex relations between preditor-prey populations and others are readily available to the population ecologist and student.

During the workshop, I realized that establishing the graphics for the various curves would not involve writing a multitude of program statements. Programs with these statements are already in place. All that are required are program lines to define what is being plotted. The graph may come from any rate dependent program, even enzyme kinetics. The use of a delete statement provides space

for the lines for a specific program required for the equation you are dealing with. What, at face view, would appear to be a very difficult and complex program is much less threatening for the beginner. The major hurdle for the novice is the writing of lines without "punctuation" or format errors.

The Biology Department owns several software programs dealing with aspects of population ecology. It seems clear that before these programs are given to a student, an explanation of the iterative power of the computer in time and number of calculations be given. This can be done by showing portions of the equations that deal with the variables of the x- and y-axis, indicating the divisions, multiplications, additions, and subtractions that make up a single point on the curve and generate the numbers for the next data point. An added bonus comes when students realize that they have dealt with these problems in calculus courses and perceive a practical application for them. Both dissolve apprehension and hopefully will lead to their exploration of computer modeling.

Existing ecology programs lack a couple of dimensions. The first seems to be that the perceptive student can rather quickly predict what a curve will look like by changing variables on the x- or y-axis. I believe that the synthesis of game theory logistics in ecology programs would provide greater student motivation. Anticipation, response, feedback, and a new response would serve to generate more sustained interest and learning by the student. I don't know whether this could be done by using existing game subroutines and inserting them into these programs. In any event, more interaction would clearly make these more valuable learning tools for the student.

2. CHEMISTRY DEPARTMENT - Rodney Olsen - "Instrument Interfacing with Microcomputers"

The goal of my activity as funded by the Exxon grant was to design and implement a computer interfaced titration experiment. It was considered essential that the system utilize equipment that was already on hand and that software be very open and understandable to any students who might be interested. The experiment which was chosen was one which has been done manually as part of Hamline's Chem 324 (Introductory Analytical Chemistry) for several years. This titration has typically been a difficult one for many students, in part because the color change at the end-point is not very sharp. For this reason it seemed to be a very appropriate choice for automation.

There are commerically available "Auto-titrators" which, in broad principle, could be used for this analytical application. These commercial units are completely self-contained and are very expensive. The software which they utilize is not written in a high-level language that can be readily examined and modified. These two serious problems make such equipment unsuitable for

teaching purposes.

Apparatus:

The computer used was an Apple II+ with an Adalab interfacing card. Adalab is a registered trademark of: Interactive Microware Incorporated, P. O. Box 771, State College, PA 16801. The computer need not be dedicated exclusively to the experiment described here and is, in fact, used extensively by the class for many other purposes as well.

The other two main components of the system were older pieces of rather conventional lab equipment. These had been under-utilized for several years and needed only to be slightly modified. Specifically, a Beckman model B spectrophotometer was modified so that a small magnetic stirrer and a 150 mL beaker could be placed in the cell compartment. (This beaker serves as the titration vessel.) A 0 to -3 volt analog output is easily connected to the Adalab's analog input. The other piece of equipment was a Sage model 249-1 motor driven syringe buret. This device was modified by the addition of a solid-state relay, which could be activated by the Adalab card, and a reflective object sensor which converts motor revolutions to pulses. These are then counted by the Adalab card.

Procedure:

The experiment is essentially the one which is described in several introductory analytical chemistry textbooks for the determination of calcium + magnesium in limestone by EDTA titration. Operational details are provided to the student in the form of printed hand-outs and messages displayed on the monitor screen when the program is run. These are available from the author as is a print-out of an actual run.

Software:

A machine language program is supplied with the Adalab card and makes its functions easily accessible from within an Applesoft BASIC program. The BASIC program which was written for this titration system interacts with the Adalab card in four ways. It acquires analog data (transmittance readings) from the spectrophotometer. It turns the syringe drive motor on and off at appropriate times. It counts pulses corresponding to revolutions of the syringe drive motor and thus acquires a measure of the volume of the titrant delivered. Also, of lesser importance, the program uses one of the Adalab's clocks to time delay periods that are necessary at certain points during the titration.

Another important aspect of the program is its function of performing calculations on the volume and transmittance data as they are acquired, and then modifying the course of the experiment accordingly. For teaching purposes, it is very desirable for the program to be written in a high-level language so that students may easily examine this function and modify it if they wish to do so.

The program was written with the intent of being "user friendly"

in a number of ways. Some of the procedural instructions are read from a text file and are presented on the monitor screen if they are needed. Default values are available for many of the user entries that are requested. The data output is formatted in an attractive and scientifically appropriate manner. The ability to save the data on a disk file for later processing, etc., is provided.

Conclusions:

The system described above was class tested during the Spring of 1985. Out of 19 students in the class, there were 13 who carried out the spectrophotometric titration as their required computer interfaced experiment. All of them had performed essentially the same experiment manually and with visual end-point detection. All of them seemed enthusiastic about having the computer take care of the drudgery of the actual titration (adding increments of titrant and recording data). Unfortunately, many of the students did not get highly accurate results. Nominally the system functioned well and most of the students indicated an understanding of what was supposed to happen. As the program was originally written, the data acquisition was inadequate during that part of the titration where the color change was occurring. The program has subsequently been modified to alleviate this problem and the system was then tested several times by one student who has consistently gotten good results. It is expected that this experiment will be used again during the Spring of 1986.

Laboratory Project Description:

A solution containing calcium and magnesium ions will be titrated with EDTA using the indicator "Calmagite." Because the quantity of indicator required is not trivial in relation to the amount of metal ions being titrated, the end-point is not as "sharp" as might be desired. A spectrophotometric observation of the color change will make possible a more reliable location of the equivalence point.

Initially the indicator is present in the form of the Mg-Ind(-1) chelate (Red, lambda max = 535 nm). As the end-point is approached, the free calcium and magnesium ions will have been "all" complexed by the EDTA. As additional EDTA is added, the Mg-Ind(-1) complex dissociates in favor of the more stable Mg-EDTA complex. At a pH of 10.0, the freed indicator anions will be predominantly in the form HInd(-2) (Blue, lambda max = 610 nm). During this period of color change, the concentration of HInd(-2) and hence the absorbance at 610 nm will increase approximately linearly as the titrant is added. As the concentration of Mg-Ind(-1) becomes reduced to near zero, the concentration of HInd(-2) ceases to increase and the absorbance remains nearly constant. Continued addition of titrant results in some decrease in absorbance because of dilution effects.

This program, SPECTRO TITRN, will record the amount of titrant added and the absorbance of the solution being titrated. Also, it will control the addition of titrant, depending on the rate of change in absorbance. After sufficient data has been taken, the data can or will be filed for later retrieval, treatment, and print-out.

The spectrophotometer essentially consists of a monochromatic light source, a sample chamber, and a light sensing device. The spectrophotometer has been modified for computer control by the addition of interfacing circuitry. The Apple-Adalab system collects data on the percent transmission and amount of titrant added and allows the computer to run a mechanical syringe. The output from the spectrophotometer's meter is fed into the system's A to D converter. The mechanical syringe is used instead of a buret to contain and deliver titrant.

Before operating the computerized spectrophotometer, read the instructions for the limestone analysis. A bottle of standardized 0.05 M EDTA will be provided, since the syringe can only hold 10 mL of titrant. Titrate 25 mL aliquots of the limestone solution used in the manual titration, adding enough water to the provided beaker for the light beam to pass through the solution. Make an estimate of the volume of the initial solution, as this will be required in the treatment of the data. A specially marked beaker may be provided. Follow the instructions in SPECTRO TITRN to initiate the titration, making sure to record the sample number.

3. ENGLISH DEPARTMENT - Michael Foote - "From Composition to Campus Word Processing"

In the Spring of 1983, I must have seemed to some an unlikely candidate as a recipient of an Exxon stipend. I had no formal background in computers, but it was a theory of mine that a microcomputer could be an invaluable aid in running the kind of not-for-profit organizations with which I had substantial experience. According, I bought Apple II+ in September 1981, and after a period of self-instruction in word processing, spreadsheet, and database, I found myself working for a chamber symphony half-time and teaching at Hamline half-time -- a temporary solution to my employment problems and impending career change.

At Hamline, having been smitten by the computer bug and firmly believing that <u>some</u> familiarity with computer keyboarding would be a requirement for employment for my students, I began to require that they write all their papers using EDT and RUNOFF on our DEC VAX-11/750 (under VMS). The results impressed me. The papers had a better look, were necessarily the product of their authors, and often even had a better content as students discovered the power of editing on a keyboard that had previously only been a typewriter. Shared problems of solving the riddles of using the often arcane commands of the system also created a camaraderie in the classroom that I had never seen before, and while a lot of the interaction concerned the proper commands, I also found that students even talked to each other about their writing and discovered strategies for improvement that might not have occurred to them in any other situation.

Perhaps it was not so unusual after all for Hamline's Exxon

Committee to offer me a stipend and suggest for my project that I distill the eight feet of DEC computer manuals into a much more modest pamphlet describing the basics of operating the system in general and using the word processing capabilities in particular.

The result was my only "bestseller", a work that has gone through two editions and something like eight printings. It has proved useful not only to my composition students, but to anyone in need of a text editor for course work or general use -- read word processing for required papers. Understanding EDT is a prerequisite for the study and use of high-level programming languages that we teach on the system, such as COBOL, Fortran, and Pascal, and this pamphlet also opened the way for those do-it-yourselfers who wanted to start in computing but had no idea how. Another salubrious effect of the appearance of this work for Hamline is that faculty started noticing the possibilities and began to require papers done on the VAX. Perhaps they discovered what I had, that such a component in a course built class spirit in ways I have seldom seen before, and from a teacher's point of view, I could alter the class message at the head of their login procedure to remind students of assignments and offer other information, and I could also ask students to mail first paragraphs of papers to each other for criticism.

Exxon's support of my work on this pamphlet has had several important and positive effects. We have now encouraged students to learn word processing to such an extent that we have overloaded the (VAX) system with it, and we are now having to find ways to take the burden of it off the VAX and accomplish this task elsewhere. Its popularity has been tremendous, and for me, it was an important exercise in learning technical writing and a major step in my development leading to a position as full-time Director of Academic Computing at Hamline (and, more recently, to a similiar position at Carleton College) -- a wonderful answer to my problem of ongoing employment. I would like to think that the money offered by the Exxon stipend was well spent both for the usefulness that the result afforded for students, faculty, and staff, and also for the development of this particular faculty member.

Exxon Software Purchase: Megaworks

At Hamline, we purchased an Apple IIe lab of 13 machines to support the many classes we offer to teachers K-12 who come to us to meet their continuing education requirements. When this lab is not used for Continuing Education, it is available for undergraduate general use and for some course instruction.

By far the most extensive undergraduate use of the Apple Lab is word processing as done on "AppleWorks." Proofreading is a difficult task to accomplish as all of us know; we often just don't see obvious typos, and then there are those times when we simply don't know how to spell a particular word even though we think we do.

At my suggestion, then, we purchased a copy of "Megaworks," a program designed to do spellchecking for any text file written in ProDOS, i.e., our "AppleWorks" text files. Students instantly

recognize the possibilities, and they have used "Megaworks" extensively to help themselves create the best copy possible. The staff in the Development Office, also, has used its mail-merge capabilities to help personalize and speed up their mailings. And I too, in order to avoid embarrassment, use it every chance I get to accomplish what I can't seem to as well in a non-technical way. The Exxon funds that provided for this relatively inexpensive improvement to our software holdings has helped a large number of people a great deal, and we are thankful for that help.

4. HISTORY DEPARTMENT

a) Richard Kagan - "Application of Computer Assisted Learning to Skills Development Courses"

[Ed. Note] The International Futures (IF) Simulation program from CONDUIT has as its purpose to introduce students to the current debate on global issues, especially in the areas of demography, economics, energy, and agriculture. Using this simulation the students can learn the substance of the issue areas, including some historic background, current problems, and forecasts. Theoretical perspectives in the issue areas include how processes are viewed and modeled in various disciplines. Options in the issue areas and political positions taken on these options by various national and international agencies are discussed. The computer programs (main program, 13 subprograms in 3,702 lines of ANSI Fortran code) represent critical domestic and international variables and relationships. With some additional work, it will be simple to use in an interactive environment, and it will allow students to test alternative policies and assumptions.

The 207-page "User's Manual" discusses the issues, their historic background, current status and forecasts; outlines theories behind the issue areas; and instructs students on how to use the model. The IF model represents the world in ten regions: the United States, Western Europe (including Yugoslavia), the rest of the western developed world (Japan, Australia, Canada, South Africa, Israel, etc.), Eastern Europe, the Soviet Union, Latin America (except OPEC members), plus the non-OPEC Middle East, OPEC, South and Southeast Asia, and China. Also included is an 82-page "Technical Manual" which documents the model's equations for more advanced users. Much work remains in order to adapt the programs for video terminals and to make the entire simulation more "user friendly."

In order to introduce my (Kagan's) students to the VAX, I began with a short course on word processing. Only three of my 30+ students had used a computer, and none of them had word processed. I had mistakenly not determined how many could type. So I began the assignment without keyboard drills.

The instructional materials were prepared by Michael Foote. The students' reaction was surprising to me. They did not like it. It was too chatty. It didn't seem clear to them. Michael kindly came in three times to help teach the material in the booklet. Nontheless, the students felt the best way to learn was to do it. The problem with that was the competition for terminals, especially during mid-term and at the end of a term, and the limited number of printers available for use. I was also not fully competent to answer all questions.

The International Futures Simulation noted above, was chosen for a historical project and was found to be not adequate for a history course. In attempting to make use of it, the following problems were encountered: (1) The program was mismatched with the monitor screen. Despite months of effort, the program failed to work adequately for the students. Only one student was able to figure it out. (2) The simulation text provided assignments that were too difficult to accomplish. (3) The text gave little or no historical context and the simulation could not be run backwards in time. (4) The simulation text, if corrected, could consume 7 weeks! It is still worth doing.

For the Spring 1985 term, I attempted to teach VAX and Apple word processing. However, because of a rather sudden change in personnel and some unexpected problems with the equipment, I was unable to implement these plans. In conclusion I offer the following: (1) A better review of existing material is necessary. (2) More terminals and printers are required. (3) Better instruction (personal and written) from the Computer Center is necessary. (4) Despite the tremendous aggravation, it was a valuable experience for both students and teacher.

b) Geoffrey Koziol - "Creation of Basic Resources"

The stipend was awarded for the "creation of basic resources," by which was envisioned the acquisition of a data bank suitable for teaching undergraduates about the use of computers by contemporary historians. Since the most frequent use of computers is for analysis of demographic data and projection of demographic trends, it was hoped that such data banks already existed. Over the summer of 1984 I located several institutions which collected and sold such data banks. The most important was the SSRC Data Archive at the University of Essex, Great Britain. Unfortunately, the data bases were not appropriate for the courses which Professor Richard Kagan and I teach. Prof. Kagan teaches East Asian history; and there were no suitable data banks on that subject. I teach early European history. The only available data banks covered modern European history; there was also a language problem in that supplementary sources to place the data in its appropriate context would have been in Latin or French, and therefore unintelligible to most Hamline undergraduates. The project remains worthwhile, but would better be addressed in courses on modern British or American history.

The stipend was also used to research the most current uses of computers in early European history. This proved quite fruitful; and the information gathered from this study was incorporated into a series of lectures and discussions in my History 113 course, "Historical Evidence and Analysis." Here students were given a general overview of the ways that historians have used computers -- and not just the usual statistical approaches but also more imaginative uses, such as computer-aided enhancements of manuscripts. Students were also introduced to a complex study of the French Revolution which used computers to analyze coded grievances which were instrumental in justifying the Revolution. Students were then given the original documents (in translation) from which the authors of the study generated their code and assigned the task of generating their own code. Both the discussion and the assignment were enormously useful not only in familiarizing students with how computers have been used, but also in teaching them that the use of computers requires a liberal application of traditional learning skills: thought and reading.

5. MANAGEMENT AND ECONOMICS DEPARTMENT

a) Chung-tai Lu - "Modification of Available Programs for Adoption in Some Economics Courses"

The overall project goal was to improve the understanding and application of economic principles through the use of computers. Computer programs have made it possible for the instructor to approach and teach, and for the students to learn and experiment with, various aspects of economics that were impossible or impracticable for the classroom or for homework. Through the use of computer programs, items that could not be covered, or covered inadequately in the classroom, can be covered or covered more adequately now, and those that could not be assigned for homework can be so assigned now. Since programs designed for many subjects in economics have been made increasingly available in recent years, it is hoped that existing programs can be selected and modified so that they fit into the specific areas of a course and become integral parts of that course. Thus with a minimal level of computer literacy, both teaching and learning can be made vastly more efficient in the sense that the students will learn more and understand what they learn better without committing more time either by the students or the instructor during the academic term.

More specifically, my personal objectives for the project period (Summer 1983) were: (1) To understand the "whys" and "hows" in the designs and constructions of selected computer programs so that I could make the modifications myself to meet exactly the needs of my courses. (2) To understand the adopted programs sufficiently so that I could communicate comfortably with my students on, and help them with the use of, these programs. (3) To search for and identify software that was most useful for my courses.

Convinced that the ability to use a computer is as indispensable as that of driving a car in a modern economy, and that I must have a machine at my disposal at all times if I was to learn to use it, I spent a good part of the summer of 1982 learning to shop for a personal computer. It was after the start of classes, however, that I finally had one delivered. Adoption of the first program for my own course, with my own computer, through my own efforts was beyond my competence and the demands of my full-time teaching load. It did not become serviceable until the following term with the help from the student assistants at the computer lab. The Exxon grant for the summer of 1983 was most timely for me -- exactly what I needed.

I was very fortunate to have met some of the student assistants at our computer lab before the summer of 1983. One of them was particularly helpful all the way; from demonstration to final checking, testing, and retesting. By the end of the summer, over 36 programs were adopted for my use. These included programs in mathematics, business mathematics, and statistics as well as in economics. Some of these duplicated somewhat the programs we already had. I did them anyway to gain more practice, for checking and testing, and to make them more suitable for my own use. Four of these programs were completely integrated into my Managerial Economics course and used there in the Fall of 1983. They are: Linear Programming, Internal Rate of Returns, Critical Path Analysis, and Program Evaluation and Review Technique. I also located some software and programs for some other courses in economics, but I was not able to try out any of them during the project period.

Our department is no longer the Department of Economics; it is now the Department of Management and Economics. I do not teach Managerial Economics any more. This course, as all other courses of the department, will be redesigned to fit into the new curriculum of the new department and into the new College of Liberal Arts curriculum. Although the place of Managerial Economics in the new departmental curriculum remains to be determined, the direction of the department is decidedly toward more computer applications. Most, if not all, of the programs adopted under this project should find more application in the developing curriculum of our department, although individual instructors may want to modify them to suit their own teaching plans.

I am gratified for the opportunity this project provided me to appreciate the power and "friendliness" of the computer. I sorely missed my computer while I was away on sabbatical in another country, and wondered often how I could have managed for decades without the word processor, the spreadsheet, and the many programs. Computers are as crucial in communications as automobiles are in transportation. With our teaching load, I found it very difficult, if not impossible, for us to improve our computer skills or to adopt programs for use in the classroom with confidence during the academic year, at least for persons with my level of computer competency. We need projects or workshops such as provided by the Exxon grant in order to acquire expertise and program resources on a regular basis. We might find ourselves stranded at the airport parking lot facing a fleet of space-age Mercedes because the only

car we can handle is a model T Ford. I am also concerned about what would happen if there were no power sources and we had forgotten how to walk.

b) Roger Walker - "Selective Computer Use in Beginning Micro Economics and Statistics Courses"

The time made available to me through the Exxon grant was invested in identifying possible alternative software, compatible with the Apple II equipment which Hamline has available for use, in both a micro-economics course and a beginning statistics course. In the case of each course, the criteria that the software needed to meet was the same: (1) the program needed to be "friendly"; (2) the substantive material covered needed to be readily integratable into the existing course; and, (3) the software needed to be such as to allow the student to realize, at least partially, the potential of computer-assisted course materials, that is, to be able to do things much faster and easier. As a result of the grant, three software packages have been identified and are to be used in these two courses during the Fall of 1985.

Biznes (CONDUIT) is a simulation package that does in micro-economics what numerous simulations do in macro-economics -- it allows students to experience a model. Just as an instructor cannot provide students with an entire economy to manage, that instructor cannot give them a real business firm to run. An instructor may, however, provide a substitute, a simulated economy or firm. This approach allows students to experiment with how concepts are related and how they interact in a theory. The 80-page "Student Manual" plays a vital role in these experiences by helping students see how the results they obtain are related to economic theory.

In Biznes, a student plays the role of a decision-maker who hires labor and capital. After the computer uses a production function to determine output, the student then chooses a price to charge. Finally, the computer prints results the student can use in completing assignments in the "Student Manual."

The primary purpose of this package is to teach students the essential concepts associated with the theory of the firm. The simulation ties together micro-economic concepts in a way that is difficult to duplicate in a textbook. The package emphasizes the importance of the three constraints that a firm faces: the demand curve, the production function, and the supply of resources. Textbooks usually combine these constraints in an effort to simplify presentations of the theory and students often do not see what is behind the scenes.

This package is designed to teach economic reasoning and not to simulate the workings of a real-world business. Use of simulation makes it easier to explain the economic theory of the firm as a limiting theory and to appreciate the importance of the simplifying

assumptions that separate this limiting theory from a theory that would replicate real-world behavior.

The instructional objectives of Biznes are: (1) Define elasticity of demand, marginal product, production function, marginal revenue, isoquant, returns to scale, and isocost line. (2) Explain the distinction between long and short run. (3) Explain what the law of diminishing returns is and under what conditions it holds. (4) Compute marginal product and approximate marginal cost when given levels of an input, output, and total cost. (5) Calculate marginal revenue and elasticity if given two price-quantity combinations. (6) Calculate marginal revenue if given price and elasticity of demand at that price. (7) Use the equimarginal principle to reach positions of minimum cost when given the marginal products and prices of two inputs. (8) Explain why the demand curve and total cost curve can be treated as boundaries. (9) List at least five assumptions that are made in the economic theory of the firm that distinguish it from the real-world firms. (10) Compute marginal revenue product from marginal product and marginal revenue. (11) Decide what changes in levels of inputs will lead to higher profits when given data on marginal resource cost and marginal revenue product. (12) Distinguish between a price maker (searcher) and a price taker. (13) If given the value of the elasticity of demand, explain what will happen to total revenue and to marginal product when output increases.

Descriptive Statistics (CONDUIT) consists of four programs on a single diskette: (A) FREQDIST - constructs a frequency distribution and graphs a frequency histogram for data in one of the three data files or for data directly entered via the keyboard. (B) FREQTAB - constructs a frequency distribution and a line graph for data in one of the three data files or for data directly entered via the keyboard. This program is not for discrete data, and it is not possible to do any grouping of data. (C) STATMEAS - computes, for either discrete or continuous data, measures of central tendency and dispersion. (D) GROUP - computes the grouped mean, variance, and standard deviation for continuous data.

This package covers the following topics: Data and Frequency Distributions (classification of data, frequency table, frequency distributions and line graphs, grouping of continuous data), Descriptive Statistics - Measures of Central Tendency (arithmetic mean - grouped and ungrouped data, median, midrange or interquartile range in a normal distribution, mode), Measures of Dispersion (range, variance and standard deviation, mean absolute deviation or the average deviation), Data Sources (three data files on diskette, keyboard entry). The three diskette files are: 100 data points - normal distribution, 250 data points - bimodal, U-shaped distribution, 365 data points - badly left skewed.

Key-Stat (Oakleaf Systems - Decorah, Iowa) introduces users to the screen format used in several other programs. Users first select from the options to ENTER DATA, LIST DATA, CORRECT DATA, SUMMARY STATISTICS, or END PROGRAM. When ENTER DATA is selected (indicated by the inverse screen) and the RETURN key is pushed, the screen clears and the choice between INDIVIDUAL VALUES and VALUES

AND THEIR FREQUENCIES (that is, grouped data) must be made. Data entry can then begin and a RETURN is pushed after each value. A -1 is used to signal that data entry is finished. The SUMMARY STATISTICS may then be displayed, which includes the sample mean, median, and total sum, plus the sample variance (VAR), sample standard deviation (S.D.), and the standard error of the mean (S.E.). The options to LIST DATA (to note item number) and CORRECT DATA can be used to replace a current value with a new value. This provides a versatile way to see the effects of extreme values on the summmary statistics. An exercise which guides students towards their own discovery of the properties of descriptive statistics can be constructed by making use of this program.

Observed frequencies are compared with expected frequencies in the "analysis of frequencies" program. The expected frequencies may be calculated from a simple ratio, a mathematical probability distribution (e.g., binomial, Poisson, normal), or be calculated from row and column totals of a contingency table. This provides the flexibility to accomodate goodness-of-fit, and R x C contingency chi-square analysis. Expected frequencies can be displayed, and users are cautioned if either any expected frequencies are less than 1 or more than 20 percent are less than 5.

In the "random sample - one population" program, random samples taken from one population can be used to compare its mean or variance to specified values. A t-score or z-score is calculated (depending on whether sample variance is used) for comparing means, and the chi-square distribution is used in comparing variances. Users are expected to become familiar with respective statistical tables (from their text or other sources) as they use this program. Options for changing data make it easy for instructors to construct "What if ... ?" scenarios for various sets of data.

In the "random sample - two populations" program, random samples are taken from two populations and can be used to compare variances, means when variances are specified, means when sample variances are used, or to compare entire distributions. The assumptions are presented for each procedure and users are referred to the F-distribution, z-distribution, t-distribution, or Mann-Whitney (= Wilcoxon Rank Sum) U-tables to determine the level of significance.

After data entry, the options for "paired data" analysis include the sign-test, the Wilcoxon signed-rank test, or the paired t-test. Users must look up significance levels in a sign table or the Wilcoxon signed-rank table. The program calculates the P-value for the t-distribution. Options for changing values, or using a different test, make this a versatile program that instructors can have students use to answer "What if ...?" scenarios for data pairs.

In the "regression/correlation analysis" program, variables are entered and a comprehensive summary table is displayed showing means, sum of squares, slope with its S.E., t-value for testing the significance of the slope, the proportion of total variation in Y accounted for by variable X, and Pearson's bi-variate correlation coefficient. A high resolution scatter plot enables users to look for any linear trends. A least squares regression line can be added

to this plot, or one of three log transformations (log Y, log X, or logs of both Y and X) are options to improve linearity and give more homogeneous variances. If there are any zero data values, a 1 can be added to each value prior to the log transformation. The ease of making data changes lets students see the effect of an extreme value on a regression or correlation analysis.

The "one-way analysis of variance" program accepts sample data from 2 to 9 different populations. Sample means and their variances are displayed along with Bartlett's statistic for homogeneity of variance. The population means can be compared, assuming equality of variance, with a calculated F-value (along with its level of significance). Users are cautioned about using the analysis of variance if population variances are unequal and the Kruskal-Wallis statistic is presented as a nonparametric option. The CORRECT SAMPLE DATA selection also provides the chance to transform data to logs which may yield more homogeneous variances.

The versatile "two-way analysis of variance" program is designed with options for a randomized complete block experimental design (that is, no replication), an R x C fixed factor factorial design (with replication), or the nonparametric Friedman test using ranks of treatments within blocks. Levels of significance are calculated for each comparison.

The first of these three software packages will be used in the beginning Micro-economics course this Fall (1985) while the other two will be used in the department's Statistics course. We plan to get regularized student reaction to the material by means of a series of questionnaires and to evaluate the effectiveness of the material at the end of the term. All three software packages seem to faculty to possess great attractiveness.

[Ed. Note] We have also used Exxon grant funds to acquire two additional micro-based statistical packages (for the Apple IIe) which have not as yet seen much use at Hamline.

A-Stat 83 (Rose Grandon Associates, 7807 Whittier Street, Tampa, FL, 33616) is distributed with four "system disks" and a 120-page "User's Guide." It includes routines for data management such as: data entry, data modification, data reporting, data exchange, data files, and facilities for missing values, alphabetic data, sorting and merging files, and transformation of variables. The following statistical routines are available: frequency distributions and histograms, cross-tabulations, correlation matrices, analysis of variance, t-tests, multiple regression, residuals and predicted scores, and factor analysis with rotations.

The WCB Statistical Packages (William C. Brown Publishers, 2460 Kerper Boulevard, Box 539, Dubuque, IA, 52001) consist of three diskettes (USTATS, MSTATS, NSTATS) plus documentation. USTATS (Univariate Statistical Treatment, Analysis and Testing System), for example, handles up to four qualitative variables with up to eight levels possible for each variable. Data manipulation features include the entry, editing, deleting, printing, transforming, and

analyzing of data. Data analysis procedures covering descriptive statistics (frequency distribution, mean, standard deviation, range, skewness, regression line, etc.) and inferential statistics (t-tests, ANOVA, trend analysis, correlation, regression, etc.) are part of this package. M = Multivariate and N = Nonparametric for the other two diskettes.

6. MODERN LANGUAGES AND LITERATURES DEPARTMENT - John Derrenberger - "Computer Assistance in Foreign Languages and Linguistics"

[Ed. Note] DASHER (CONDUIT - mainframe BASIC version, Apple version also available) - can be used to create language drills in a question and answer format for elementary and intermediate students. The instructor must specify one correct answer for each question. The instructor turned author needs only a minimal understanding of computers and BASIC; the program handles all of the technical aspects of building the drills. DASHER uses a unique graphical method for reporting students' errors by displaying visual clues. This program is available to other disciplines where formulas or lists of items constitute correct answers. Included is a 150-page "Instructor's Manual" for this 2,622-line BASIC program (with lots of REMarks).

The purposes I (Derrenberger) undertook to fulfill with the stipend from Exxon in the summer of 1984 were twofold: (1) to write the code for a driver to the DASHER subroutines and to create data files for this new software based on the introductory German language text, Deutsch Heute. (2) to evaluate the usefulness of computer assistance in elementary foreign language instruction, with particular reference to German.

As regards my programming activities using the DASHER program, I can be brief. After DASHER arrived from Iowa and after a bit of fiddling (most admirably accomplished by Gary Anderson), the program was loaded onto the Hamline VAX.

Brimming with enthusiasm for the project, I tested out the DASHER program and found that it fulfilled its promise of analyzing character strings, comparing them to correct strings included in the program either as DATA statements or as data files, and marking any points of disagreement in a reasonably intuitive manner. All of which is to say, the DASHER routines held the promise of being useful in foreign language studies by examining student input, comparing this input to an instructor's/author's version, marking any errors as appropriate, and waiting to examine corrections.

I began work on writing the code for an Index or Driver program for the many subroutines provided with the main program. I chose additionally to write a Driver which would access data files as opposed to DATA statements. (n.b., the DASHER routines are all in BASIC.) This turned out to be a very difficult task. I am not an

experienced programmer. I found it difficult and tedious to keep track of all the many variables included in the main program and to call them up as needed in a useful fashion.

The arrival of the regular term and its attendant business relegated the DASHER work to the fourth tier. Furthermore, early in the Spring term 1985 I learned that the author of the DASHER program, James Pusack of the University of Iowa, was writing the Driver and the data files to achieve precisely the goals I had sought. He was even basing the new program on the same textbook, Deutsch Heute. Needless to say, I simply decided to buy what Professor Pusack would produce.

The other part of my mission was to evaluate the applicability of computers to the teaching of elementary German. My comments in this regard are based on Pusack's DASHER drills for Deutsch Heute and the TUCO program developed at Ohio State.

Because we have not yet instituted a program of computer assistance in elementary German, I cannot comment on such matters as effects on student motivation and speed and accuracy of learning.

I can, however, make the following observations. Since language is overwhelmingly a spoken, i.e., oral, event, the probability that computers will play a significant role in the main lines of language instruction in the proximate future is slim. There is no need here for a treatise on the complexity of natural human language, but surely all know that language is an incredibly complex phenomenon -- so complex indeed that linguists have yet to write a complete and accurate grammar of any language.

By this reasoning it is clear that computer assistance cannot address linguistic complexity with primitive programs and microcomputers. What computers can accomplish, in my view, is this: Computer programs can assist students in the areas of inflectional morphology by providing rewarding feedback and infinite patience in drills. Computer programs can assist students to learn vocabulary symbols, i.e., spelling, in a variety of entertaining ways. It should be noted, however, that motivated students readily understand that a language does not exist apart from its vocabulary and are, therefore, usually quite eager to learn vocabulary at any rate. Computer programs do not seem to be well suited to such other tasks as syntactical modifications and emulations of conversation, because of the enormous number of variables which must be accounted for in the programming.

7. PHILOSOPHY DEPARTMENT - Joseph Uemura - "The Use of Symbolic Logic Systems for Teaching Logic"

In the summer of 1984, I received an Exxon grant in response to my proposal having the title listed above. The procedure I followed in this development project was rather close to what I thought to be

helpful at the time: (1) to find out the relevant resources, (2) to familiarize myself with some of the techniques, and (3) to make some observations concerning the feasible use of such resources and techniques in the classroom.

In order to find out the relevant resources, I have consulted various libraries and commercial outlets concerning available software in the teaching of logic -- as they call it, CAI, i.e., Computer Assisted Instruction, in logic. This is largely a futile venture, because such systems are only beginning to appear in any real commercial sense.

Taking a lead from Project Director Gary Anderson, I began writing to various places which are known to have teaching-by-computer courses in their curricular announcements. So, I wrote to: (1) Dartmouth College (for BERTIE), (2) Ohio State, (3) Carnegie-Mellon, and (4) Pennsylvania. The most fruitful response came from Professor Preston Covey who recently published an article on "Computer-Assisted Instruction in Philosophy" in the Journal of the Modern Language Association. Most valuable for our use, here, is the list he gives of the micro-computer courses in logic instruction.

At the suggestion of Professor Ron Messerich, who taught a summer (1984) course at Hamline on the History and Philosophy of the Computer, I proceeded to research other courses. Actually, I also heard from a former Philosophy student of mine, who was in the process of producing such a course, Christopher Driesbach; and as a friendly gesture he sent me a copy of his course. This is the course called ALICE. Oddly enough, it is, I found out subsequently, probably the best course currently available. It was recently "auditioned" at the Western Division meeting of the American Philosophical Association in Chicago. So, I am saying that I discovered this resource in this rather serendipitous manner. We have, now, two copies of ALICE -- one purchased with Exxon funds.

Responding to my request for information, Dartmouth's Professor James Moor sent me information concerning the symbolic logic system which has become popular there called BERTIE (after Bertrand Russell, of course), and also said that I could have a copy of the VAX-tape they use if I desired it. I did, of course, and we now have it on the Hamline VAX system.

As to books in computer-assisted instruction in logic, I have reviewed many. They are beginning to multiply like a veritable fungus. As a sample: Schagrin, Rapaport, and Dipert. They will become increasingly popular and very available.

My familiarization with techniques, as you can imagine, has been usual, elementary, and modest. I have worked with the Director of Academic Computing on the VAX, and I have worked with ALICE on the Apple IIe, and I have worked with William Zinsser's book, Writing with a Word Processor, for my own amusement. Our departmental assistant, Mr. Gordon Wax, has become proficient in use of the Cavalier-Driesbach course, ALICE.

Just to make some observations about CAI in logic, after trying out these things, I would make some very tentative observations, based on only a modest amount of experience:

1. I think, certainly, Logic in its elementary and symbolic forms can certainly be taught in this way. It is a relatively simple business in the first place, and a relatively mechanical business, too. One might feel a bit that it is "too mechanical" and "too simple," as well. But, together with all of the other feelings about computers, such feelings are pretty common in any subject-matter.

2. Specific to Logic, as such, it seems to me that the procedure of memorization is somewhat inadequate in learning by the computer. It does not seem, to learn Logic visually, that the process is quite repetitive enough, practiced enough, or something like that, in itself. It is my judgment that the process needs to be augmented with considerable amounts of concurrent writing, drawing of Venn diagrams, rewriting obverses, converses, and the like, before the memory grasps just what is going on. So, there needs to be a lot of pencil-and-paper work to supplement the computer-work itself.

3. This, too, might be said, about talking things over with someone -- at least, a class-mate; perhaps, a teacher. Many of the moves seem to be "arbitrary and mechanical." Logic, already, seems to be arbitrary and mechanical enough, in the best of all possible Logic classes. This exacerbates that process and needs to be countered by conversation, dialectic, perhaps outright arguments.

8. POLITICAL SCIENCE DEPARTMENT - John Harrigan - "Increasing Computer Research Facility Among Political Science Students"

I received an Exxon grant to develop a project to increase the usage of computers in Political Science instruction. Specifically, my goal was to teach students in my Political Science 354 (Introduction to Political Research) how to use SPSS programs in conducting political science data analysis.

I approached this task with some trepedation, because my experiences with Hamline's VAX system up to that time had been mostly negative. I had made some use of the VAX for word processing, but had been recurringly put off by the computer being down for long periods of time when I needed to do some work (e.g., the famous lightning strikes), the printers not working when I was under a deadline to run off something, difficulty getting access to a terminal because there were so few of them and so many people needing access, and computer technicians with whom I could not communicate very well. In frustration, I eventually purchased my own personal computer.

Up to this time I had also used the VAX periodically to run various SPSS programs for my own research. And in this area my

frustrations were even worse. My biggest problem was that the computer operators would periodically change the operating commands needed to access SPSS, run the programs, or get printout. Needless to say, these changes were seldom communicated in advance to the VAX users. Being a computer neophyte myself, I found these changes very difficult to handle.

So you can probably understand my trepedation when I approached the task of devising a means for having 35-40 students using the VAX under these same conditions. Nevertheless, familiarity with SPSS or comparable statistical packages is extremely important in Political Science, and I wanted my students to have that familiarity.

So, upon receiving the Exxon grant, I spent the better part of the summer of 1983 drafting detailed instructions and models for students to use with SPSS. Our data base was the National Opinion Research Center General Social Surveys for the years 1973-1982. My set of instructions gave each student a model for using the NORC data base to test hypotheses, principally using the "Crosstabs" program and the chi-square test for statistical significance for two-way cross tabulations. Then the exercise got them into using three-way cross tabulations to experiment with the impact of control variables. By the end of the summer I had pretested this model several times and was convinced that it was in perfect working order and ready to be implemented with the class. Upon completion of that task, I began preparing similar models for the SPSS programs in correlation analysis and regression analysis.

When the Fall 1983 term started, I began to encounter what for me were insurmountable problems with the people who run the VAX. The first problem was a long delay in getting student accounts set up. Despite repeated requests (that started out politely and ended up being stated with exasperation) the Fall term was one-third over before the student accounts were set up. By this time the system manager was so angry at me that he refused to talk to me. When the accounts were finally set up, the system manager then changed the commands necessary to access SPSS, run the programs, and get output. This one act (a system improvement?) totally negated all the work I had done the previous summer. The instructions I had prepared and distributed to the students were now almost totally useless. In frustration, I cancelled the project for that Fall's class. I bought my own computer for my own purposes, and I haven't logged on to the VAX since then.

Even though I loathe the idea of having anything to do with the VAX under the conditions I have experienced in the past, I may reconsider my personal abstention. Recent personnel changes, the possibility of getting more terminals, the acquisition of SPSS-X and the Political Science Department's purchase of the 1985 version of NORC's General Social Survey maybe will make it easier for computer users who are non-specialists. If so, I will be delighted to expand usage of the VAX in my classes. If not, I will design exercises that can be carried out on the Apple microcomputers on campus, or I will simply leave computer-assisted instruction to other people at Hamline who are much more patient and more adept at it than I am.

9. PSYCHOLOGY DEPARTMENT

a) Charles LaBounty - "Essay Testing by Computer in a PSI Class"

Developmental Psychology at Hamline is taught using a modified personalized system of instruction (PSI). For each of the 19 units of text and lecture material, the student is provided with a list of essay questions to use as a study guide. Depending on the length and complexity of the unit, the questions may number from a dozen or so to over 30. When upon feeling confident in a particular unit, a student may take a test that consists of five questions chosen at random from the study guide list. Any students receiving an unsatisfactory grade on a unit test may retake another form of the test.

In the past, giving tests by paper forms meant the preparation of many different forms of each test in order to prevent students from getting identical tests on retakes and to prevent students from informing others what was on a particular unit exam. It was decided to attempt the unit testing by means of computers.

In concept, the procedure was rather simple. Students are given study guides for each unit. When they are ready to take a test, they come to the Apple computer laboratory during one of several weekly testing periods. They request the appropriate unit, and a student assistant hands them the unit disk and their personal testing disk. The computer then presents a sequence of five questions randomly selected from the total list and collects the student's typed answer (up to 17 lines). After the student finishes, the instructor or student assistant reads the disk, scores the answers, and types appropriate comments after each answer. When the student takes the next test, prior to its presentation of the questions, student answers, and reader's comments of the previous exam are displayed on the terminal screen.

During the Fall 1984 and Spring 1985 terms, a total of 83 students were exposed to computer testing using this system. Success rate, as measured by drop-out rate, was comparable to previous years with no computer use. Success rate, as measured by average grade achieved, was slightly higher than in previous years with no computer use.

At the end of each term, students were polled for their opinions of computer testing. Reactions were mixed. Most enjoyed the novelty of testing on the computer. Many said that it was their first time at a terminal and found it a relatively non-threatening introduction. A few even commented that they enjoyed the typing practice.

Several serious negative comments were made, however. With only a dozen terminals available, students often had to wait for long periods of time to gain access to a machine. This was particularly true toward the end of each term when the twin problems of procrastination and desire for make-up exams added to the pressure.

Further, those students with no, or minimal, typing skills found it took much longer to complete an exam on the keyboard than they would need using pencil and paper.

The student assistants and the instructor also identified another problem. It would take approximately five minutes to read and react to a single student exam. A comparable paper and pencil test usually took less than two minutes. This additional time, when multiplied by 83 students times 19 exams, was considerable.

It has not been decided whether to continue computer testing in this class. Three improvements might help. First, if more micros were available the user crunch would be reduced. Second, if enrollment limits were set lower, demand for machines would be less, as would be the test scoring workload. Finally, additional funds for more student assistants would help by enabling more laboratory hours and more time for reading exams. If any two of these improvements were enabled, the instructor would be encouraged to continue the program.

b) R. Kim Guenther - "Computers in Cognition"

Cognitive Psychology is a course which examines perception, attention, memory, problem solving, learning, and intelligence. The modern perspective on human cognition is called information processing -- this perspective likens thinking to a series of stages which manipulate symbolic information. Information processing models are often studied in experimental paradigms which require very precise timing of behavior. Information processing also has a natural metaphor in the programs of computers -- both information processing models and computer programs are algorithms for making decisions on the basis of symbolic activity. Computers, then, are useful for testing the experimental predictions of information processing and for simulating cognitive activity.

I required students in my course in Cognitive Psychology to participate in a variety of some of the "classic" information processing experiments and to examine computer simulation of cognition-like natural language comprehension. Some of the experiments were run by two software packages purchased from CONDUIT (Laboratory in Cognition and Perception; Computer Lab in Memory and Cognition -- see IV-9 of this report). From these packages I used Pattern Interpretation, Levels of Processing, Semantic Memory, Prose Comprehension, Comprehending Inferences, and Reasoning from Prose. In addition, I developed my own software to run an experiment in retrieval from short term memory, an experiment in learning by elaboration, and a series of experiments investigating a specialist view of intelligence. All of the software was written in Applesoft BASIC and run on an Apple IIe computer. Students were able to check out diskettes from the library and run them themselves at convenient times. The programs presented the instructions, provided the graphics necessary for several of the experiments, and measured reaction time to millisecond accuracy for those experiments

requiring speeded responses.

For each experiment, students were required to do some background reading, run themselves in the experiment, record their results, and answer a set of questions about the experiment's purposes, methodologies, and implications. Nearly all the students regarded participation in the experiments as valuable and interesting. The experience helped students understand research methodology in Cognitive Psychology and heighten their interest in the subject matter.

The computer was also used to help familiarize students with recent attempts to simulate intelligence. Specifically, the students observed the ELIZA program (written in Applesoft BASIC) which simulates the natural language processing of a client-centered therapist. This demonstration helped students understand the mechanisms and limitations of computer simulation and provided a springboard for comparing human and artificial intelligence. For example, computer programs can easily be written to simulate rote memorization but not so easily written to simulate gist memory. Yet humans can readily remember the gist of a message but only with much effort rote memorize all of a message's details.

I intend to continue to require, in this class, participation in a set of information processing experiments although I may add a few more experiments to the lab. I also hope to expand demonstrations and discussion of artificial intelligence.

c) Matthew Olson - "Computer Assisted PSI Testing"

Since the 1975-76 academic year, the Psychology Department of Hamline University has provided PSI (Personalized Style of Instruction) options for many of its courses. Students unable to take courses in the traditional mode have found the self-paced, mastery-oriented PSI plan appealing, and faculty have avoided problems encountered when enrollments become unreasonably large. PSI utility has been demonstrated most convincingly in the General Psychology (introductory) course where traditional and PSI sections each have enrollments of 70-100 students per term.

Benefits of the PSI plan do not come without associated costs, however. In a typical term, three to five student proctors are hired to perform supervision, test-distribution, and test grading/recording functions for the General Psychology course alone. Two to three additional proctors manage similar functions in upper level PSI courses. As it is currently structured, proctors provide a number of designated testing times, distribute exams as requested by students, collect and grade completed tests, and record results. At the end of the term, proctors tally final results and submit the records to course instructors for final grade approval.

Implementation of a 1983 Exxon grant makes a gross reduction in proctor function feasible and provides for a more personalized and

even more dependable examination format. The Psychology Department TEST package (designed by Matthew Olson and written by Paul and Lisa Angstman) is an extremely flexible testing package written in Applesoft BASIC and used on a departmental Apple IIe with two disk drives. It is user friendly, provides input error checks, and should be relatively tamper-proof. The complete package consists of a number of programs to be used by an instructor/supervisor and a single program for student use.

Instructor Programs

MASTERFILL is a program which allows the instructor to create a multiple-choice question pool of a specified size for each chapter or unit to be tested. The program prompts for question entry, for four multiple choice options, and for correct response.

EDIT is a program which displays questions exactly as they appear to students during testing. It allows for any sort of correction and rewrites the correct item to the question pool. EDIT is the only program which allows access to pool items once they are written by MASTERFILL.

NAMEFILL is a program which creates a file for each participating student. Students are identified by name and a three-digit code.

ADDNAME is an appendix program which creates files for students who add the class after the initial file set has been created by NAMEFILL.

GRADE is a program which displays scores from single files on a per-unit basis. It also keeps a running total of examination points and an overall percentage score.

Student Program

TEST is the only program to which students (should) have access. Given a RUN TEST input, the program prompts for name, identification number, and test unit. In its current form, TEST allows two attempts at a given unit exam and saves the larger of the two exam scores. It does not allow the student to retake material from unit "N" once the student has accessed a test from unit "N+1," even if the student has taken only one exam from unit "N."

TEST takes a random sample (of some predetermined size) from items in the pool created by MASTERFILL and displays them sequentially with prompts for correct response. It provides immediate feedback on a per item basis, including corrective information when the student makes an error. The program does not provide the option of deferring a given item and returning to it after other test items have been attempted.

Upon completion of a given exam, TEST records the total number of correct answers in the file created by NAMEFILL (or ADDNAME) and allows the student to continue accessing examinations or to terminate.

EVALUATION ---

Student Response

In the Spring term of 1984, the Computer Assisted Testing (CAT) version of PSI testing for General Psychology was offered to students on a voluntary basis. Students who chose to participate took the first 8 of 16 examinations in the standard paper-pencil format and completed the remaining 8 exams using the TEST program. Only seven students chose to participate so actual comparison of ratings for course effectiveness between the CAT students and the 60+ students in the paper-pencil group would have been statistically meaningless. All participating students stated that they preferred the computer mode to the standard PSI mode. Similarly, all students felt that the immediate correction and feedback capability made CAT superior to the standard grading procedure. Although three of the seven participants expressed a desire for implementation of a deferral option, they also agreed that the immediacy of feedback in some way compensated for deferral of difficult items.

After reworking and improving the original TEST program, CAT testing appeared as part of the the published course schedule for Fall term 1984. (Note that reworking was addressed to better machine-student interaction and testing speed. Deferral of test items was not made part of TEST capability.) Only five students enrolled in the CAT section, and three of those students -- all freshmen -- transferred to the standard section when it became clear that testing was in the CAT mode. The two remaining participants, both upper-level students, expressed the same satisfaction indicated by the first group of students. Neither of these students expressed the need for the deferral option. The CAT section of PSI General Psychology and the supervising instructor were both on Sabbatical during the Spring term of 1985.

Instructor Response

Small enrollments in the CAT sections have been disappointing and probably reflect the same "machine-aversion" observed by instructors in the Computer Literacy programs. It is interesting to note that all successful CAT students except one were upper-level students who had previous computer experience in the college.

Despite enrollment problems, TEST clearly serves the function for which it is designed. Supervision of testing is reduced to a single introductory session in which students are assigned identification numbers and instructed in the initiation of the program. Additional instructor time is optional. (For example, this instructor chose to make weekly copies of the student files to avoid accidental information loss.) Determination of final grades is greatly simplified, and the element of human error is minimized.

The system does suffer two major problems. The first lies in availability. There is only one testing station, and that microcomputer is also used for a number of other Psychology Department functions. The Bush Library Apple Lab has alleviated some user congestion, however, and there are plans to acquire more

departmental machines late in 1985. A second problem lies in the use of MASTERFILL. The program requires that an operator enter items by hand, and the process can be quite time-consuming. While many publishing companies provide Apple-compatible test-banks, MASTERFILL is not capable of accessing those items directly from company disks and writing them into pools for use by TEST.

Future Plans -- TEST will be used extensively during the Fall term 1985. The instructor will be teaching a small traditional section of General Psychology. Anticipated enrollment is 35 students and all objective testing will be done via TEST. The substantial feedback derived from that group of students should provide informative directions for further refinements and use of the package.

10. RELIGION DEPARTMENT - Walter Benjamin - "Cost/Benefit Paradigms Applied to Biomedical Ethics"

I used my Exxon stipend to contact a number of health care delivery corporations and systems to discover the utilization in diagnosis, distributing effective care, and the like. They eschewed my probing questions concerning any "cost/benefit" paradigms that they might be employing. However, as a WCCO-TV "I Team" discovered last Fall with its research into HMOs, there are procedures which, beause of expense and rarity, are not covered.

I met with health professionals at Health Central Incorporated, Group Health Incorporated, several Twin Cities hospitals, Blue Cross/Blue Shield, etc. Because of oversupply, many physicians are turning out software which will help emergency room diagnosis and the like. Medical schools, such as Yale, are also writing programs. Solo practitioners in remote areas also have computer linkages with medical schools and medical attendings at tertiary centers to reduce diagnostic and medical regimen unknowns and mistakes.

11. SOCIOLOGY DEPARTMENT

a) Peter Falkman - "From SPSS to SPSS-X / Differences and Similarities"

The purpose of this project was to become familiar with new software features of SPSS-X (Statistical Package for the Social Sciences). This software package uses a batch system for analyzing and displaying statistical data in the Social Sciences.

I had become familiar with SPSS in graduate school and was teaching a data analysis component in the Sociology Department's Senior Seminar, using SPSS. In 1983, an updated version of the

program (SPSS-X) came on the market and the Hamline Computer Center made the switch to only running SPSS-X. Thus, if I was to continue using this package in our department's course offering and in my research, I felt I should become familiar with this new variation.

During the summer of 1984, I received a stipend from the Exxon grant to prepare myself with the new instructions and commands of SPSS-X. Three goals were set: (1) to become familiar with the new commands and to allow input of data directly into the SPSS command file; (2) to set up SPSS-X program commands and bring in data that was already on file (the National Opinion Research Council - NORC - data files); and (3) to set up a separate data file and a SPSS-X command file and to call both together to allow for analysis of the data. During that summer, goals 1 and 2 were accomplished. Goal 3 was accomplished during the Fall term 1984 with a great deal of trial and error.

To accomplish my goals, it was necessary to read and become familiar with the following publications, manuals, and technical reports: SPSS Combined Edition - complete documentation for SPSS batch system through release 9; SPSS Primer; SPSS-X User's Guide - operational documentation for first release of SPSS-X; SPSS-X Basics - begins with the mechanics of submitting SPSS-X jobs and continues through file definition, data transformation, tabulation, and report writing. For each of the three goals, the objective was to be able to output descriptive statistics, cross-tabulations, scattergrams, correlations, and multiple regression procedures using SPSS-X.

As a result of the summer work, I decided to offer an SPSS-X course in the January 1985 Interim (a four-week course devoted to a single topic). The course went well with twelve students enrolled from the departments of: Economics, Psychology, Sociology, Political Science, and Mathematics. Students performed statistical analysis using the NORC data files.

I requested that the Department purchase a U. S. Census data tape (the 5% Public Use Sample) and this was ordered in the Spring of 1985. Plans are to use this data with SPSS-X in a Senior course in Social Research Methods. A course in the use of SPSS-X was offered during a 1985 Summer Session. Students in the Master of Public Adminsitration program at Hamline have expressed an interest in such a course in order to analyze data for their Master's theses.

b) David Tavakoli - "Computer Assisted Instruction in Socioloc,"

This is a brief report of my computer-related activities throughout the Summer of 1984 and the academic year 1984-85. In the Summer of 1984 I received a Bush grant to attend a computer workshop, "Using the Computer in Teaching Sociology," at Ball State University in Muncie, Indiana. This workshop was sponsored by the American Sociological Association for Sociology instructors. This was not my first contact with computers, as I had learned to use both large mainframes and microcomputers in my research during my

Ph.D. training at the University of Minnesota. At the Muncie workshop, the participants were introduced to several different uses of both mainframes (Ball State University has a VAX-11/780 that is larger than Hamline's VAX-11/750) and microcomputers in sociology instruction. Topics of discussion included: "Computer Literacy and the Sociology Curriculum," "Large Screen Graphics Demonstration," "Microcomputers and Computer Literacy," "Developing Computer Literacy: How to Make the Machine Work for You," "Integrating SCSS-GSS Data," "Microcomputers: Spreadsheets and Statistics," "SCSS-Straight Forward," "Survey-Research Data Base Instruction," "Microcomputers: CONDUIT and Random House Simulations," "Hands-on Experiments with an Assortment of Microcomputers in the Lab," "SPSS-X on the Mainframe," "Using the Mainframe with Simulations and Computer-Assisted Communications: CONDUIT and Inter-University Linkages," "Microcomputer SPSS-X," "Simulations: CONDUIT -- Mainframe," and a wrap up session on exchanging ideas by the participants using an assortment of microcomputers.

In the following academic year, 1984-85, I was awarded an Exxon stipend to continue my study of the available software for instruction in sociology. I used this grant to attend a major conference in San Antonio, Texas, where a follow-up workshop to Muncie's was held, attended mainly by the same people who were in Muncie. I also took a class during the Fall quarter 1984 at the University of Minnesota to reacquaint myself with SPSS and SPSS-X and recent methods in data analysis. This course was taught by a sociologist who is an expert in the application of computers in sociology and has systematically used computers in sociology instruction. After returning from Muncie, and throughout the year, I have shared with my colleagues in the Department of Sociology what I had learned in these workshops. Professor Margaret Jensen used "Computers in Crime" (CDC micro-PLATO courseware) in her Criminology class. Other colleagues have looked over these materials. The Exxon grant thus helped to generate quite a number of computer-related activities in my own sociology classes.

During the Fall 1984 term, I began to use several of these software packages in my sociology classes. I was interested in finding out undergraduate students' perception of the utility of these packages for instruction. To this end, I devised and used a one-page evaluation form in assessing students' opinions of these packages. After completion of each experiment, I asked students to fill out the evaluation form. These data are now being analyzed in a study evaluating the available software in sociology instruction in conjunction with similar data collected at the University of Minnesota by Professor Ronald Anderson and at Inver Hills Community College by Professor Cid Jackson. Copies of my paper acknowledging Exxon's grant will be available after the completion of this study.

As I developed my interest in computer-related activities, I approached the Director of Teaching Services of the American Sociological Association, who had organized both the Muncie and San Antonio workshops, to consider Hamline University as a site for a similar workshop. What I had in mind was to get some local and national visibility for Hamline University and to bring additional computer resources and expertise to Hamline. I was not surprised to

find out that some of the participants in these workshops were aware of the fact that Hamline University is the first institution of higher education in Minnesota with a computer literacy graduation requirement. As a result of my initiation, the American Sociological Association held a similar workshop at Hamline University, June 20-23, 1985, in which 17 sociology instructors from across the United States participated.

At HU Workshop IV (4/18/85), I presented to a selected number of faculty the initial results of my evaluation of the available software in sociology. I learned a great deal from other colleagues, some of whom have gone far beyond the phase of software collection and are beginning to develop special programs for their own courses. My hope is to continue using what I have learned so far in my sociology classes during the next academic year.

In conjunction with the activities outlined above, I have used part of the Exxon grant to purchase a number of software packages that are now available at Hamline for students and faculty to use. A complete inventory of these materials is available and can be obtained from the Project Director.

IV. HARDWARE / SOFTWARE / COURSEWARE ACQUISITION

Hardware/software/courseware acquisition as a grant activity has proved to be most helpful in moving Hamline towards its goal of "computing as a matter of course." Departmental budgets seldom had any funds allocated for software acquisition, so it was with great pleasure that the Project Director was able to solicit suggestions for such purposes. The major hardware purchases of the grant were to acquire a University of Delaware Sound Synthesizer for use with our CDC 110 micros and the Micro GUIDO Ear-Training System and a secondary disk drive for a CDC 110 micro. All other funds for this activity were used for software/courseware acquisition.

Before discussing the aquisition and use of software in this grant activity, we will briefly describe our current academic computing facilities. In the summer of 1982, Hamline installed a DEC VAX-11/750 computer system with 3 million characters of main memory and 500 million characters of disk storage. Both primary and secondary storage capacity have been increased since that time. This system supports 30 intelligent graphics terminals (DEC-GIGI) plus several graphics and letter-quality printers which are located in three campus (cluster) locations. Our VAX has many high-level programming languages (Ada, BASIC, COBOL, Fortran, Pascal), several word processing systems, DATATRIEVE, SPSS-X, electronic spreadsheets, RS/1, SLATEC, CAS, plus a complete set of graphics support programs and electronic mail. Hamline has a number of assorted microcomputers for special applications that are located in various departments, e.g., DEC PDP-11/23 under RSX with a 20 megabye disk, DEC LSI-11's, two Tektronix 4051 high resolution graphics computers, several Apple IIe's, IBM PC's, and a number of single board micros.

Bush Memorial Library currently houses two microcomputer laboratories. A dozen Apple IIe systems were purchased by Continuing Education and are used by undergraduates when not booked for CE courses. Hamline has three IST III terminals (host PLATO) plus six CDC 110 micros, the latter being used mainly for micro-PLATO courseware and CP/M software.

Administrative computing will be moving back on campus in 1985 with the addition of another VAX-11/750. Plans call for adding many DEC Rainbow micros to serve as stand-alone units and/or terminals to either of the VAX systems. As can be expected, facilities and staff are in a state of continual change as Hamline struggles to meet future computing needs. We find ourselves to be quite software "rich" but, in a sense, user "poor". Hence an immediate challenge is to increase the academic computing user base so as to realize the goal of increasing at least the incidental use of computing in many disciplines.

Perhaps the foremost problem in the acquisition of course-related software is that of determining what is available, useful, and compatible with the HU computing systems. Several copies of CONDUIT catalogs were circulated to various faculty members and this

generated considerable interest and several large orders. Software reviews (e.g., CHOICE) and discipline-oriented publications and meetings provide another source for this type of needed information.

Acquiring what we hope to be useful software is only the first step. Implementing the use of this software is the next, and probably the most difficult, problem. It's just not like buying a new toaster that can be made operational by unpacking it and plugging it in. Software implementation is highly labor intensive and time consuming. We expect to be in this phase of computer utilization long after the grant period has expired. Many programs have to be adapted to a specific version of the high-level language that they were written in (assuming source code was purchased). Other program adaptations include screen formatting, creation and use of (temporary) output files, and other somewhat technical changes. These changes need to be made in close consultation with the end users (faculty). Technical staff (currently in short supply at HU and elsewhere) need to be aware of the faculty users' needs and to exercise a good deal of patience in working with the faculty. The documentation that accompanies some of the packaged programs is sometimes very "intimidating" and hence needs to be rewritten in a shorter and more "user friendly" mode. These are necessary activities, but they are also very time consuming. The Project Director served to coordinate most of these activities and secured purchase orders, paid invoices, distributed software, etc. Both faculty, staff, and student help will need to be utilized in this implementation phase.

After software has been acquired and adequately adapted to a specific computer system, the implementation process needs to be directed towards classroom use. The faculty member in charge of the course that will be using the adapted software will have to decide where, when, and how such software will be used. The following reports are in addition to those of the previous section which may have addressed the issues of software acquisition and implementation. The reports in this section are quite brief, do not cover all of the grant-funded acquisitions, sometime speak to work in progress, and are designed to give the reader some idea of the problems and practice involved in such activities. Also included are occasional comments from the Project Director (and editor). Please contact the individual faculty member noted for additional information relating to a specific software package.

1. ANTHROPOLOGY DEPARTMENT - Cynthia Cone

"Cultural Evolution: A Simulation for Use in Introductory Anthropology and Sociology Courses" - Michael Davis, Dee Medley and Gene Muehlbauer, Lake Forest College, 5350 Glenwood Ave., Chicago, IL, 60640.

In the Spring of 1985, the Hamline University Anthropology Department introduced the use of a computer game to further the basic understanding of one theory of cultural evolution. The

program was set up as non-mandatory work in which participants each received 10 extra credit points for successfully running through the program. This entailed moving a society from a nomadic hunting and gathering mode of subsistence to a society at the level of a centralized state political organization through choosing among a number of solutions for a series of problems the society faced.

The "game" takes a student approximately 30 minutes to run through twice. The initial run of the program was with the assistance of a proctor. The proctor introduced the program as one designed according to the theory of cultural evolution proposed by Julian Steward.

The proctor provided some guidance in discussing with the students the underlying rationale for their choices. Students subsequently ran the program alone until their score achieved the minimum acceptable for extra credit.

The simulation allowed students to "put into operation" a particular theoretical approach to evolution in order to experience the effects of changes and their implications for the future of a society. The game was successful enough to be instituted on a required basis and the department has ordered an updated version of it.

2. CHEMISTRY DEPARTMENT - Clifford Creswell

The following CONDUIT simulations were acquired: (1) Chem Lab Simulation #1 - acid-base titration, Avagadro's number, weak acid equilibrium constant, pH meter simulation; (2) Chem Lab Simulation #2 - ideal gas law, kinetic molecular theory, diffusion; (3) Chem Lab Simulation #3 - Hess's law of heats of reaction, calorimetry; (4) Chem Lab Simuilation #4 - thermodynamics, heat of vaporization, enthalpy, entropy, free energy; (5) Reaction Kinetics - chemical reaction kinetics, first and second order reactions, rate constants, concentrations.

The software noted above is well done, especially the graphics. However, the programs will find limited use in our chemistry program. We have the students perform experiments in the laboratory on titrations, calorimetry, thermodynamics, and kinetics. I find the programs most useful in generating data quickly in class to illustrate some principle or relationship. They could be used by students in a pre-laboratory experience. However, having a limited number of copies makes this difficult for all students to use.

I especially liked the programs illustrating the gas laws, the kinetic molecular theory, and the principles of diffusion. These programs were very effective in introducing the concepts in class. We will continue to use these programs in that capacity.

3. COMPUTER SCIENCE PROGRAM - Gary Anderson

The micro-PLATO educational software packages described here are primarily used in CS 100 - Computer Fundamentals - to provide instruction that covers some of the social issues in computing. Diskettes with CDC 110 operating instructions are placed on library reserve. This courseware is from the Control Data "Computer Literacy Series" and is also available for the Apple II and IBM PC. Most of these courses are used as supplemental assignments in a more comprehensive approach to the topics discussed. We also have several other micro-PLATO courseware packages available at Hamline (Introduction to Computers, Computers in the Classroom, Programming in MBASIC, Lower Division Engineering Curriculum (LDEC) courses: General Chemistry, General Physics, Fortran Programming, etc.).

These micro-PLATO courses are formatted like many computer-assisted tutorials. Short segments are followed by a series of multiple-choice questions with immediate feedback telling whether the answer was right or wrong and a short sentence giving additional information about the correct and wrong choices. Several graphics figures are used as well as some role-playing exercises to lengthen the student's attention span. The screen size and data rate serve to limit the amount of information that can be presented at one time. Student reaction has been rather wide, ranging all the way from boredom to excitement. Although supposedly at the college level, some courses seem to be more at the junior high level. Each courseware module includes a 10-12 page booklet. Following are some brief descriptions of this courseware currently in use at Hamline.

Artificial Intelligence (3-4 hours) - introduces the use of artificial intelligence to mimic selected aspects of human intellectual functions. Pattern recognition, natural language processing, frame theory, adaptive capabilities, and problem solving are graphically depicted, defined, and reviewed with various learning activities.

Computer Crime (2-3 hours) - reviews the impact of computer crime on business, personal finances, and society in general.

Computer Privacy (3 hours) - introduces the concept of personal information access through electronic networks. Privacy considerations and concerns are presented, with exercises on U.S. privacy laws and possible precautions against computer-based violations of personal privacy.

Computers and Employment (2-3 hours) - is designed to present an overview of the social implications of computer automation for business and the workforce. Various job situations are explored and computer capabilities are explained.

Computers and Government (3 hours) - describes how the computer has changed how the government functions. Applications include the use of the computer by the IRS, the military, the Census Bureau, and Congress. Other current and future applications address the impact of the computer on the ability of the government to store and

maintain information.

Computers and the Future (1 hour) - simulates the decision-making on a space ship in a hypothetical future society. This module notes the wide variety of possible new computer technologies in the future. The social and ethical implications of some of these new computer uses are highlighted.

Electronic Funds Transfer (2-3 hours) - introduces the process of electronic funds transfer as it relates to the corporate world as well as the private sector. Present and future applictions, the impact on the banking and finance industry, and potential weaknesses are addressed.

Human Factors and Computers (3 hours) - addresses the factors which affect the interface between humankind and the computer. Positive and negative affects of hardware and software design and human reaction to these designs are reviewed.

Robotics and Computers (3 hours) - presents the concept of robotics, a topic related to artificial intelligence. The present and future applications of robots in industry are reviewed along with an interesting simulation in which the student "programs" a robot to do a simple task.

Social Values and the Computer (1 hour) - is a simulation-based exercise for exploring the psychological, social, and economic impact of the computer on individuals, organizations, and society.

4. EDUCATION DEPARTMENT - George Redman

Project Description

In order to provide technologically up-to-date programs for Hamline students in teacher education, the Education Department designed and implemented an Exxon-funded project to acquire new and quality software for use in teacher education courses and programs. It was intended that the software could be used to enrich the following kinds of courses: elementary methods, secondary methods (ED 575), required courses (e.g., ED 562 - Education and Cultural Diversity).

In addition, it was intended that the software programs acquired could be utilized by college faculty outside of the Education Department and by Education faculty who teach in the continuing education program for teachers.

Finally, any materials that would relate to "computer literacy" could be integrated into the CS 100 (Computer Fundamentals) course. In short, the software acquired would be utilized by faculty and students within the Education Department as well as by faculty and students outside of the Department. Thus the bulk of the activity related to the project centered around ordering, previewing, and

selecting software for purchase. A partial listing of such items is attached to this report.

Evaluation of Progress

The review and selection phase of the project has been successfully completed. All materials have arrived and have been catalogued for faculty and student use.

What we could call "phase two" -- the actual integration of the software into courses -- has already begun also. For example, the "Grade Manager" program was demonstrated in ED 575, a general methods course for teachers. Several of the elementary-level programs have been utilized in the elementary methods course sequence (ED 550 and ED 555). Some of the Logo materials were utilized in CS 100. Lately, all of the materials were utilized in ED 501CE (Computer Literacy for Teachers) during the Spring term of 1985. I anticipate that these types of uses will increase as faculty and students become more aware of the availability of these programs.

In summary, the Exxon funds have made available new and enriching materials to Education faculty and students as well as to faculty and students in other departments. Students are beginning to benefit from the integration of these programs into courses. I believe that this project is indeed beginning to make an impact on the level of computer literacy on the Hamline campus.

Following is an abbreviated list of some of the materials obtained from MECC (Minnesota Educational Computing Consortium): Apple New Users Guide, Using Computers in the Classroom, Applesoft Special Features Diskette and Support Manual, Apple Assembly Language Booklet, Grade Manager, Teacher Utilities, Data Handler Support Booklet, Food Facts Support Booklet, Introduction to Logo for Teachers, Master Spell, Discovery Lab, Computers in Teaching.

5. HEALTH AND PHYSICAL EDUCATION DEPARTMENT - Richard Tressel

The Champs/0 and Champs/D football scouting programs were used by this department in a variety of ways. Physical Education majors must complete a coaching practicum for state coaching certification. The students doing their practicum in football coaching found their scouting experience using the Champs programs very beneficial and enlightening. They had practical experiences in the application of these programs to real situations.

The programs were also demonstrated in our Theory of Football course for Physical Education majors as an example of current coaching aids. The Champs program uses in the intercollegiate (MIAC) football program were obvious. As the Physical Education faculty grows in its understanding of this software, many more students in the coaching theory classes will use these programs.

The connection of computer usage to activity and sport was perhaps more important to our department and its majors. We must continue to apply modern technology to our discipline. The students will then have a foundation and awareness for possible future application of the computer in Physical Education.

6. MANAGEMENT AND ECONOMICS DEPARTMENT - [Editor's Note]

The following two CONDUIT packages have not as yet been subjected to any implementation efforts. Much work needs to be done to adapt these programs to the VAX and to make them more interactive and easy-to-use. We expect that such implementation will occur as student independent study projects in the near future. After that has been accomplished, a considerable amount of faculty (and student) time and effort will be necessary to enable these packages to become useful in the classroom.

Business Management Laboratory is a (Fortran) computer simulation of several decision-making functions found in the stainless steel flatware industry. It covers both the essentials of the industry and of simulated individual firms within the industry. The simulation can be varied in complexity by the administrator and thus is adaptable to various situations. Firms are managed by student teams who replace previous management. Each team competes for the product market, starting from essentially the same position with the same product. Included are a 74-page "Participant's Manual" and a 316-page "Administrator's Manual."

The Executive Game is a (Fortran) simulation of an imaginary business environment in which participants can practice the art and science of planning and decision-making at the top management level. Managers must deal simultaneously with decisions in production, marketing, finance, and competition. Teams make decisions on price of product, marketing budget, research and development budget, production volume scheduled, investment in plant and equipment, purchases of materials, and dividends declared. Included are a 189-page "Student Manual" and a 44-page "Instructor's Manual."

7. MATHEMATICS DEPARTMENT - Gary Anderson

POLYMATH (Control Data) is a menu-driven computational system that allows a user to apply numerical analysis techniques to small-scale problems. The system combines current state-of-the-art numerical algorithms with the interactive and graphic capabilities of the CDC 110 microcomputer. POLYMATH automatically plots virtually any function and provides outputs in the form of graphs, curves, tables, and charts. All instructions, messages, and other communications are presented in English and/or math symbols so that no programming knowledge is needed. This package includes the following routines: simultaneous linear and nonlinear algebraic

equations solver, simultaneous linear and nonlinear differential equations solver, polynomial linear and nonlinear curve fitting (with user-defined transformation functions), multiple variable linear regression, and a general purpose calculator capability.

POLYMATH has a demonstration section with easily understandable example problems to help students learn its use. There is a HELP key. Students found some of the routines rather cumbersome to use and execution times quite slow. In response to student suggestions, a short user's guide will be written in order to make better use of this interesting package.

SuperCalc (Sorcim - CDC) is an electronic spreadsheet program designed for CP/M-based microcomputers (CDC 110). Words and numbers are displayed in a worksheet format. Status information and prompt messages on the screen assist the user in setting up the worksheet by step-by-step keystrokes. Various "help" messages are available at all times. Data can be changed easily and numbers recalculated quickly. Once worksheets are generated, they can be stored on diskette to access, edit, and printout later. One CDC 110 micro is connected to a graphics printer for hardcopy output.

In addition to a printed "User's Guide and Reference Manual" which is included with the program diskette, there are many other publications relating to SuperCalc use. Since our main use of SuperCalc has been for independent study projects, we also have "Getting Started with SuperCalc", a CDC micro-PLATO course (CDC 110) which has four lessons (45 minutes each). These lessons are: SuperCalc Power, Using an Existing Report, Creating a Report Skeleton, and Evaluating Formulas. Other electronic spreadsheets are available on other HU computer systems. It is expected that this type of spreadsheet will soon be introduced in appropriate courses at Hamline.

Ordinary Differential Equations (CONDUIT) by John L. Van Iwaarden for the Apple IIe (and adapted for the VAX) will be used in conjunction with his new textbook (Harcourt Brace Jovanovich, 1985) for MA 572 - Differential Equations with Numerical Methods. The programs in this package are menu-driven, include many numerical methods, and provide for high-resolution graphics. Each program has an "equation generating" file associated with it and new algorithms (in Applesoft BASIC) can be added. These programs plus the use of POLYMATH and SLATEC (on the VAX) will be used in the Fall 1985 version of MA 572.

8. MUSIC DEPARTMENT - George Chu

[Ed. Note] GUIDO is an acronym for Graded Units for Interactive Dictation Operations and is also the name for Guido D'Arezzo, an eleventh century monk who invented the staff and the solfeggio symbols do, re me, fa, sol, and la. The Micro-GUIDO Ear Training

system consists of a CDC 110 microprocessor and a University of Delaware Sound Synthesizer plus an 8" diskette that contains the courseware. With complete programming in intervals, melodies, chords, harmonies, and rhythms, the GUIDO system uses high resolution graphics, touch input, and a fully programmable sound synthesizer to provide a rich music learning environment. Fifty-six hours of dictation exercises are available along with an orchestration program that students can use to change timbres. A comprehensive set of instructor options allows teachers to adjust the Micro-GUIDO ear training system to their own needs.

The system was used conjunction with the Music Theory I-II-III curriculum. Some difficulties were encountered in operating the equipment, but these have apparently been remedied. We also found it somewhat difficult to use the Micro-GUIDO system directly in the Theory Courses. To be sure, there were certain students who benefited greatly by having this resource available for drill and practice, but in general our students seemed to prefer working with each other at the keyboard. Naturally, this all may have reflected the instructors' somewhat restricted use of such as tools as well as the students' previous experience.

As for future use, we will continue to experiment with Micro-GUIDO use in these courses. At a small college with small classes, these types of programs seem less relevant than at a large institution where instructor-student ratios are unfavorable. If one of our Music faculty had the time and expertise to write programs directly for our courses, this equipment could be put to greater use.

9. PSYCHOLOGY DEPARTMENT - R. Kim Guenther

Topics in Research Methodology: Main Effects and Interaction

This software package from CONDUIT introduces students to factorial designs, helps students determine if main effects or interactions are present by using graphic techniques, provides a social psychology experiment with which to test students' knowledge of main and interaction effects, and describes some higher order factorial designs. Graphics and the interactive quality of the software make this package a useful way to learn about factorial designs. The package was made an option for students in an advanced experimental design class (Research Design and Method). Several students worked through several sections of the package and claimed the lessons helped them better understand the concept of interaction. I intend to make the software required "reading" in future versions of Research Design and Method.

Human Factors and Computers

This software package from Control Data's PLATO computer-based

education series includes three diskettes introducing students to human factor considerations in workstation, software, and hardware design. Students in Computers in Psychology were required to work through the software design lesson which illustrated human factors principles in the use of function keys, screen displays, sequence control, and data entry. The assignment required students to turn in a worksheet on which they listed some of the human factors principles discussed in the lesson (e.g., the program should anticipate and make allowances for common misspellings and abbreviations). The students generally regarded this exercise as helpful. I intend to continue to use this software (Apple IIe and CDC 110 micros) in future installments of the Computers in Psychology course.

Artificial Intelligence (AI)

This software package from Control Data's PLATO computer-based education series includes six diskettes that cover a variety of topics in AI including: an introduction to the field, representation, search, problem solving, expert systems, and future developments. The programs use graphics and interactive techniques that require students to answer questions. The expert system identifies mammals from user-supplied data and so simulates expert systems used in medical diagnoses. This set of diskettes was optional "reading" for students in Computers in Psychology. These students were required to write a paper on AI. Some of the students used this software as reference material and were generally enthusiastic about it. I intend to continue to use this package (Apple IIe and CDC 110 micros) in future installments of Computers in Psychology and possibly in my Cognitive Psychology class as well.

General comment: We need a better system for making back-up copies of these sorts of courseware diskettes if they are not copy protected. One diskette is too few for larger courses and any damage to the diskette permanently eliminates the lesson. Perhaps a site license-to-copy agreement should be made part of any such purchase contract.

S U M M A R Y

This report thus includes most of the activities of the Hamline University Computer Literacy Faculty Development Project. This project involved faculty from many academic departments and encouraged them to investigate the possible use of computers in their courses. Without the funding made available by the Exxon Education Foundation, such progress in making "computing as a matter of course" would not have been possible.